University of Moratuwa

Faculty of Engineering

Department of Electronic and Telecommunication Engineering

EN3251 - Internet of Things

Laboratory Exercise 1: MQTT Implementation and Testing

Group 10

210325M – Kuruppu M.P.

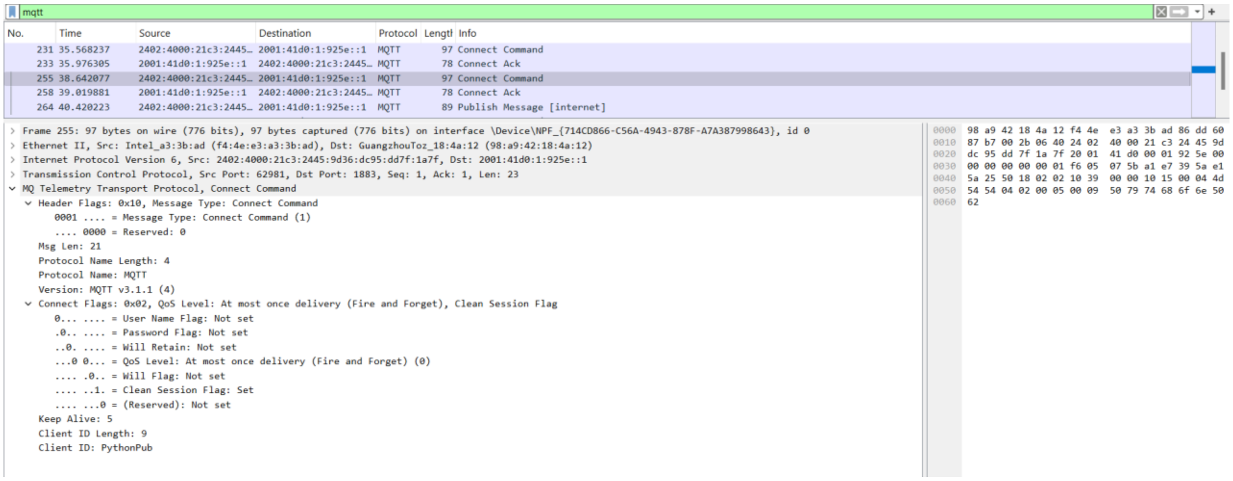
210454G – Peiris D.L.C.J.

210463H – Perera L.C.S.

Step 2

Computer A: Publisher

Connect Command

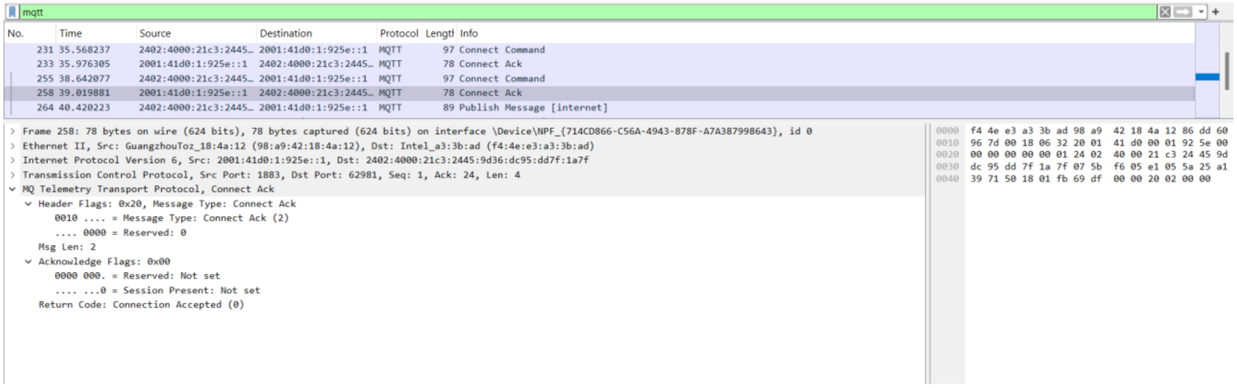


```
mqtt
No.    Time       Source                Destination           Protocol Lengtl Info
   231 35.568237  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        97 Connect Command
   233 35.976305  2001:41d0:1:925e::1  2402:4000:21c3:2445… MQTT        78 Connect Ack
   255 38.642077  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        97 Connect Command
   258 39.019881  2001:41d0:1:925e::1  2402:4000:21c3:2445… MQTT        78 Connect Ack
   264 40.420223  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        89 Publish Message [internet]
```

```
> Frame 255: 97 bytes on wire (776 bits), 97 bytes captured (776 bits) on interface \Device\NPF_{714CD866-C56A-4943-878F-A7A387998643}, id 0
> Ethernet II, Src: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad), Dst: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12)
> Internet Protocol Version 6, Src: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 62981, Dst Port: 1883, Seq: 1, Ack: 1, Len: 23
v MQ Telemetry Transport Protocol, Connect Command
  v Header Flags: 0x10, Message Type: Connect Command
      0001 .... = Message Type: Connect Command (1)
      .... 0000 = Reserved: 0
    Msg Len: 21
    Protocol Name Length: 4
    Protocol Name: MQTT
    Version: MQTT v3.1.1 (4)
  v Connect Flags: 0x02, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
      0... .... = User Name Flag: Not set
      .0.. .... = Password Flag: Not set
      ..0. .... = Will Retain: Not set
      ...0 0... = QoS Level: At most once delivery (Fire and Forget) (0)
      .... .0.. = Will Flag: Not set
      .... ..1. = Clean Session Flag: Set
      .... ...0 = (Reserved): Not set
    Keep Alive: 5
    Client ID Length: 9
    Client ID: PythonPub
```

```
0000  98 a9 42 18 4a 12 f4 4e  e3 a3 3b ad 86 dd 60
0010  87 b7 00 2b 06 40 24 02  40 00 21 c3 24 45 9d
0020  dc 95 dd 7f 1a 7f 20 01  41 d0 00 01 92 5e 00
0030  00 00 00 00 00 01 f6 05  07 5b a1 e7 39 5a e1
0040  5a 25 50 18 02 02 10 39  00 00 10 15 00 04 4d
0050  54 54 04 02 00 05 00 09  50 79 74 68 6f 6e 50
0060  62
```

Connect ACK:



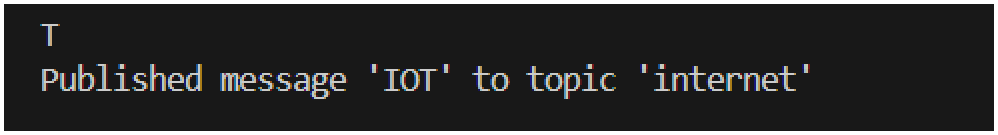
```
mqtt
No.    Time       Source                Destination           Protocol Lengtl Info
   231 35.568237  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        97 Connect Command
   233 35.976305  2001:41d0:1:925e::1  2402:4000:21c3:2445… MQTT        78 Connect Ack
   255 38.642077  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        97 Connect Command
   258 39.019881  2001:41d0:1:925e::1  2402:4000:21c3:2445… MQTT        78 Connect Ack
   264 40.420223  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT        89 Publish Message [internet]
```

```
> Frame 258: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{714CD866-C56A-4943-878F-A7A387998643}, id 0
> Ethernet II, Src: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12), Dst: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f
> Transmission Control Protocol, Src Port: 1883, Dst Port: 62981, Seq: 1, Ack: 24, Len: 4
v MQ Telemetry Transport Protocol, Connect Ack
  v Header Flags: 0x20, Message Type: Connect Ack
      0010 .... = Message Type: Connect Ack (2)
      .... 0000 = Reserved: 0
    Msg Len: 2
  v Acknowledge Flags: 0x00
      0000 000. = Reserved: Not set
      .... ...0 = Session Present: Not set
    Return Code: Connection Accepted (0)
```

```
0000  f4 4e e3 a3 3b ad 98 a9  42 18 4a 12 86 dd 60
0010  96 7d 00 18 06 32 20 01  41 d0 00 01 92 5e 00
0020  00 00 00 00 00 01 24 02  40 00 21 c3 24 45 9d
0030  dc 95 dd 7f 1a 7f 07 5b  f6 05 e1 05 5a 25 a1
0040  39 71 50 18 01 fb 69 df  00 00 20 02 00 00
```

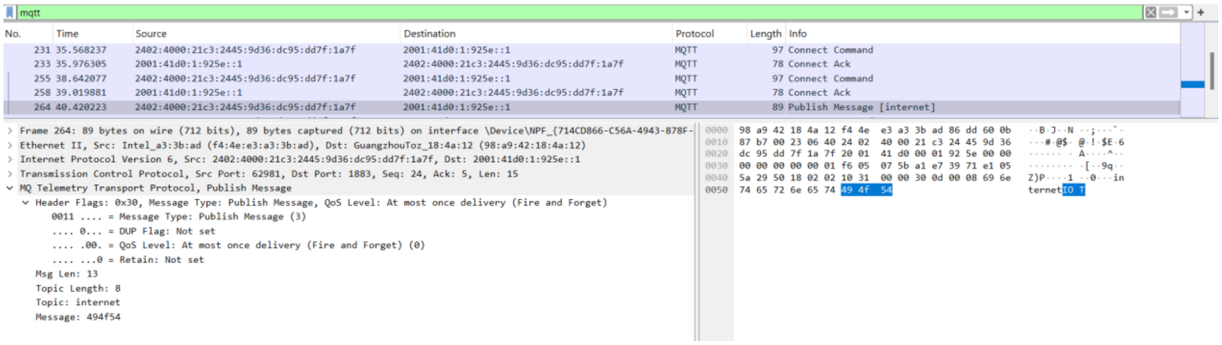
Publish Message:

Published Message from the Publisher's Side.



```
T
Published message 'IOT' to topic 'internet'
```

Wireshark Packet Capturing from the Publisher's Side



```
mqtt
No.   Time       Source                                   Destination                              Protocol  Length Info
  231 35.568237  2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f  2001:41d0:1:925e::1                      MQTT          97 Connect Command
  233 35.976305  2001:41d0:1:925e::1                      2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f  MQTT          78 Connect Ack
  255 38.642077  2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f  2001:41d0:1:925e::1                      MQTT          97 Connect Command
  258 39.019881  2001:41d0:1:925e::1                      2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f  MQTT          78 Connect Ack
  264 40.420223  2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f  2001:41d0:1:925e::1                      MQTT          89 Publish Message [internet]
```

```
> Frame 264: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device\NPF_{714CD866-C56A-4943-878F-
> Ethernet II, Src: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad), Dst: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12)
> Internet Protocol Version 6, Src: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 62981, Dst Port: 1883, Seq: 24, Ack: 5, Len: 15
v MQ Telemetry Transport Protocol, Publish Message
  v Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)
      0011 .... = Message Type: Publish Message (3)
      .... 0... = DUP Flag: Not set
      .... .00. = QoS Level: At most once delivery (Fire and Forget) (0)
      .... ...0 = Retain: Not set
    Msg Len: 13
    Topic Length: 8
    Topic: internet
    Message: 494f54
```

```
0000  98 a9 42 18 4a 12 f4 4e  e3 a3 3b ad 86 dd 60 0b   ··B·J··N ··;···`·
0010  87 b7 00 23 06 40 24 02  40 00 21 c3 24 45 9d 36   ···#·@$· @·!·$E·6
0020  dc 95 dd 7f 1a 7f 20 01  41 d0 00 01 92 5e 00 00   ······ · A····^··
0030  00 00 00 00 00 01 f6 05  07 5b a1 e7 39 71 e1 05   ········ ·[··9q··
0040  5a 29 50 18 02 02 10 31  00 00 30 0d 00 08 69 6e   Z)P····1 ··0···in
0050  74 65 72 6e 65 74 49 4f  54                        ternetIO T
```

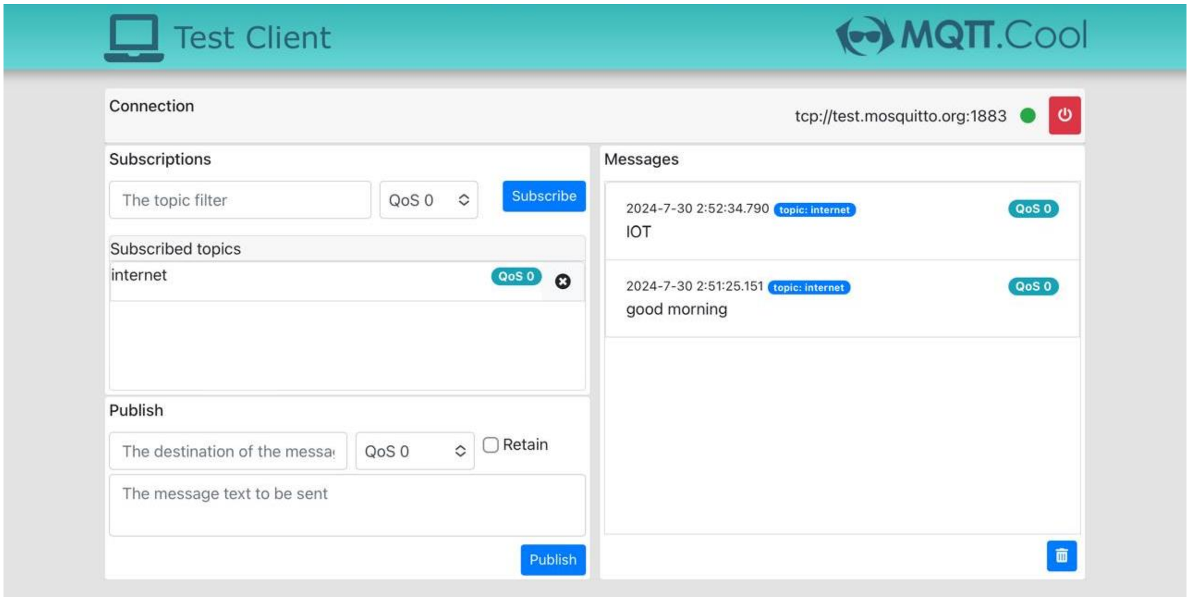
Connect Command: Computer A (the publisher) initiates a connection to the MQTT broker by sending a Connect Command using TCP port 1883.

Connect Acknowledgment (ACK): The MQTT broker responds with a Connect Acknowledgment, confirming that the connection has been successfully established.

Publish Message: Computer A sends a Publish Message to the MQTT broker. The message is intended for the topic "internet." The payload content of the message is "IOT," as decoded from the hexadecimal sequence 494f54. The packet details indicate that the QoS level is 0, which corresponds to an "at most once" delivery—no acknowledgment or retry is expected from the broker.

Computer C: Broker

MQTT Subscriber and MQTT.Cool Test Client Configuration - Monitoring message receipt and topic subscriptions during the MQTT communication test.



Connection

tcp://test.mosquitto.org:1883 ●

Subscriptions

The topic filter | QoS 0 | Subscribe

Subscribed topics

internet QoS 0

Messages

2024-7-30 2:52:34.790 topic: internet QoS 0
IOT

2024-7-30 2:51:25.151 topic: internet QoS 0
good morning

Publish

The destination of the messa | QoS 0 | □ Retain

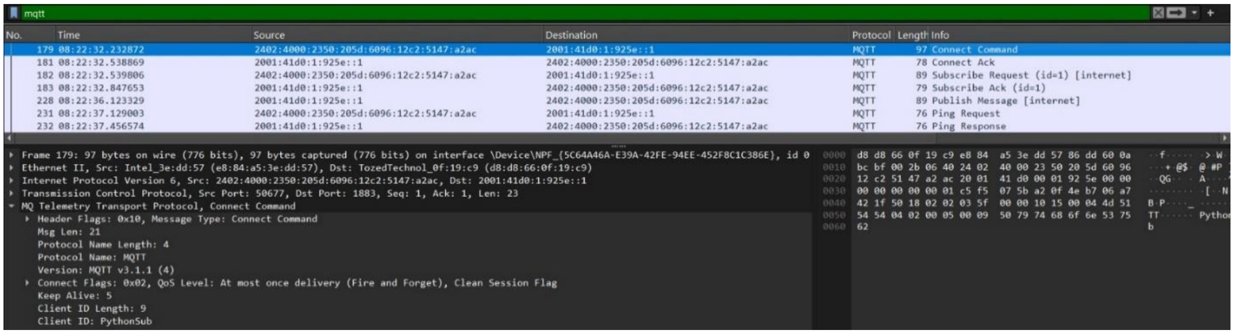
The message text to be sent

Publish

Both Publisher and Subscriber are connected via tcp://test.mosquitto.org:1883, monitoring message receipt and topic subscriptions during the MQTT communication test

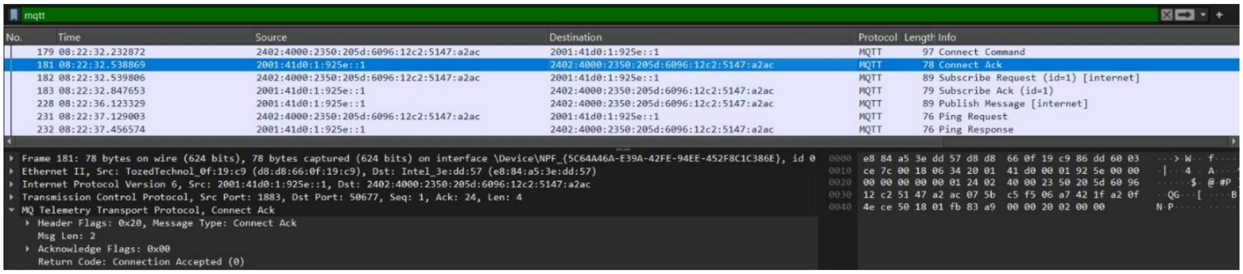
Computer B: Subscriber

Connect Command:



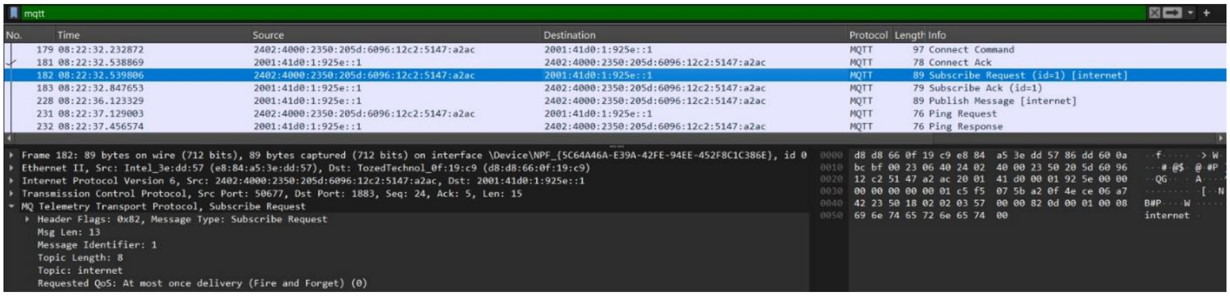
```
mqtt
No.  Time             Source                                   Destination                              Protocol Length Info
 179 08:22:32.232872  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     97 Connect Command
 181 08:22:32.538869  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Connect Ack
 182 08:22:32.539806  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     89 Subscribe Request (id=1) [internet]
 183 08:22:32.847653  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     79 Subscribe Ack (id=1)
 228 08:22:36.123329  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     89 Publish Message [internet]
 231 08:22:37.129003  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
 232 08:22:37.456574  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response

> Frame 179: 97 bytes on wire (776 bits), 97 bytes captured (776 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, id 0
> Ethernet II, Src: Intel_3e:dd:57 (e8:84:a5:3e:dd:57), Dst: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9)
> Internet Protocol Version 6, Src: 2402:4000:2350:205d:6096:12c2:5147:a2ac, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 50677, Dst Port: 1883, Seq: 1, Ack: 1, Len: 23
v MQ Telemetry Transport Protocol, Connect Command
  > Header Flags: 0x10, Message Type: Connect Command
    Msg Len: 21
    Protocol Name Length: 4
    Protocol Name: MQTT
    Version: MQTT v3.1.1 (4)
  > Connect Flags: 0x02, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
    Keep Alive: 5
    Client ID Length: 9
    Client ID: PythonSub

0000  d8 d8 66 0f 19 c9 e8 84  a5 3e dd 57 86 dd 60 0a   ··f·····  ·>·W·
0010  bc bf 00 2b 06 40 24 02  40 00 23 50 20 5d 60 96   ···+·@$·  @·#P
0020  12 c2 51 47 a2 ac 20 01  41 d0 00 01 92 5e 00 00   ··QG····  A···
0030  00 00 00 00 00 01 c5 f5  07 5b a2 0f 4e b7 06 a7   ········  ·[··N
0040  42 1f 50 18 02 02 03 5f  00 00 10 15 00 04 4d 51   B·P····_  ······
0050  54 54 04 02 00 05 00 09  50 79 74 68 6f 6e 53 75   TT······  Pythor
0060  62                                                 b
```

Connect ACK:



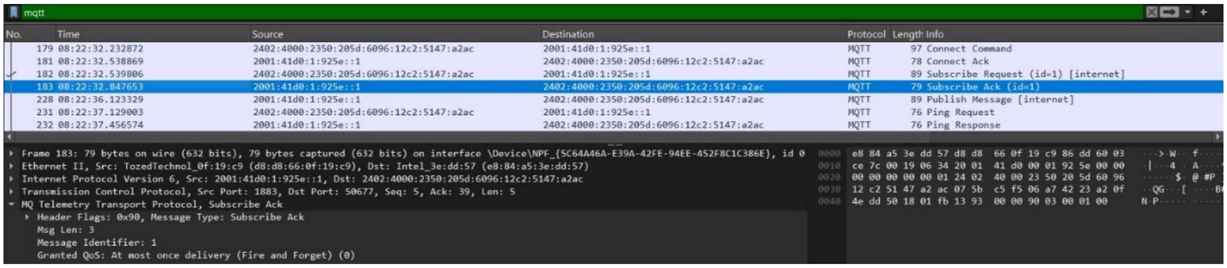
```
mqtt
No.  Time             Source                                   Destination                              Protocol Length Info
 179 08:22:32.232872  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     97 Connect Command
 181 08:22:32.538869  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Connect Ack
 182 08:22:32.539806  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     89 Subscribe Request (id=1) [internet]
 183 08:22:32.847653  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     79 Subscribe Ack (id=1)
 228 08:22:36.123329  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     89 Publish Message [internet]
 231 08:22:37.129003  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
 232 08:22:37.456574  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response

> Frame 181: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, id 0
> Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
> Transmission Control Protocol, Src Port: 1883, Dst Port: 50677, Seq: 1, Ack: 24, Len: 4
v MQ Telemetry Transport Protocol, Connect Ack
  > Header Flags: 0x20, Message Type: Connect Ack
    Msg Len: 2
  > Acknowledge Flags: 0x00
    Return Code: Connection Accepted (0)

0000  e8 84 a5 3e dd 57 d8 d8  66 0f 19 c9 86 dd 60 03   ···>·W··  f····
0010  ce 7c 00 18 06 34 20 01  41 d0 00 01 92 5e 00 00   ·|···4··  A····
0020  00 00 00 00 00 01 24 02  40 00 23 50 20 5d 60 96   ······$·  @·#P
0030  12 c2 51 47 a2 ac 07 5b  c5 f5 06 a7 42 1f a2 0f   ··QG···[  ····B
0040  4e ce 50 18 01 fb 83 a9  00 00 20 02 00 00         N·P·····  ······
```

Subscribe Request:



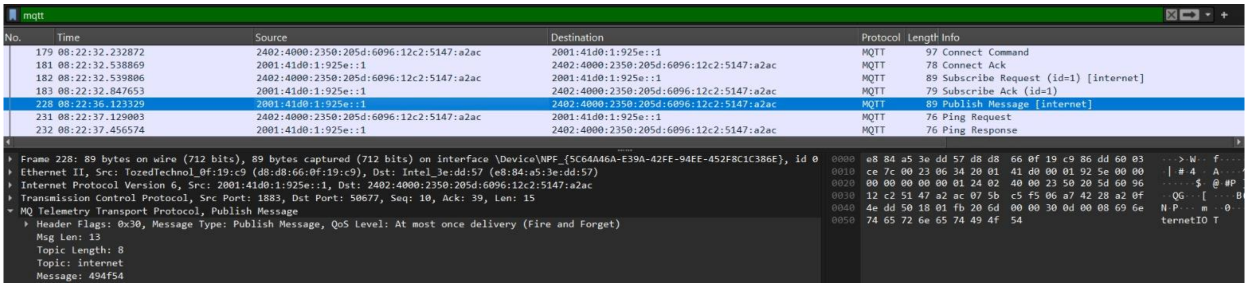
```
mqtt
No.  Time             Source                                   Destination                              Protocol Length Info
 179 08:22:32.232872  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     97 Connect Command
 181 08:22:32.538869  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Connect Ack
 182 08:22:32.539806  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     89 Subscribe Request (id=1) [internet]
 183 08:22:32.847653  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     79 Subscribe Ack (id=1)
 228 08:22:36.123329  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     89 Publish Message [internet]
 231 08:22:37.129003  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
 232 08:22:37.456574  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response

> Frame 182: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, id 0
> Ethernet II, Src: Intel_3e:dd:57 (e8:84:a5:3e:dd:57), Dst: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9)
> Internet Protocol Version 6, Src: 2402:4000:2350:205d:6096:12c2:5147:a2ac, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 50677, Dst Port: 1883, Seq: 24, Ack: 5, Len: 15
v MQ Telemetry Transport Protocol, Subscribe Request
  > Header Flags: 0x82, Message Type: Subscribe Request
    Msg Len: 13
    Message Identifier: 1
    Topic Length: 8
    Topic: internet
    Requested QoS: At most once delivery (Fire and Forget) (0)

0000  d8 d8 66 0f 19 c9 e8 84  a5 3e dd 57 86 dd 60 0a   ··f·····  ·>·W·
0010  bc bf 00 23 06 40 24 02  40 00 23 50 20 5d 60 96   ···#·@$·  @·#P
0020  12 c2 51 47 a2 ac 20 01  41 d0 00 01 92 5e 00 00   ··QG····  A···
0030  00 00 00 00 00 01 c5 f5  07 5b a2 0f 4e ce 06 a7   ········  ·[··N
0040  42 23 50 18 02 02 03 57  00 00 82 0d 00 01 00 08   B#P····W  ······
0050  69 6e 74 65 72 6e 65 74  00                        internet  ·
```

Subscriber Ack:



```
mqtt
No.  Time             Source                                   Destination                              Protocol Length Info
 179 08:22:32.232872  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     97 Connect Command
 181 08:22:32.538869  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Connect Ack
 182 08:22:32.539806  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     89 Subscribe Request (id=1) [internet]
 183 08:22:32.847653  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     79 Subscribe Ack (id=1)
 228 08:22:36.123329  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     89 Publish Message [internet]
 231 08:22:37.129003  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
 232 08:22:37.456574  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response

> Frame 183: 79 bytes on wire (632 bits), 79 bytes captured (632 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, id 0
> Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
> Transmission Control Protocol, Src Port: 1883, Dst Port: 50677, Seq: 5, Ack: 39, Len: 5
v MQ Telemetry Transport Protocol, Subscribe Ack
  > Header Flags: 0x90, Message Type: Subscribe Ack
    Msg Len: 3
    Message Identifier: 1
    Granted QoS: At most once delivery (Fire and Forget) (0)

0000  e8 84 a5 3e dd 57 d8 d8  66 0f 19 c9 86 dd 60 03   ···>·W··  f····
0010  ce 7c 00 19 06 34 20 01  41 d0 00 01 92 5e 00 00   ·|···4··  A····
0020  00 00 00 00 00 01 24 02  40 00 23 50 20 5d 60 96   ······$·  @·#P
0030  12 c2 51 47 a2 ac 07 5b  c5 f5 06 a7 42 23 a2 0f   ··QG···[  ····B
0040  4e dd 50 18 01 fb 13 93  00 00 90 03 00 01 00      N·P·····  ·····
```

Publish Message:



No.	Time	Source	Destination	Protocol	Length	Info
179	08:22:32.232872	2402:4000:2350:205d:6096:12c2:5147:a2ac	2001:41d0:1:925e::1	MQTT	97	Connect Command
181	08:22:32.538869	2001:41d0:1:925e::1	2402:4000:2350:205d:6096:12c2:5147:a2ac	MQTT	78	Connect Ack
182	08:22:32.539806	2402:4000:2350:205d:6096:12c2:5147:a2ac	2001:41d0:1:925e::1	MQTT	89	Subscribe Request (id=1) [internet]
183	08:22:32.847653	2001:41d0:1:925e::1	2402:4000:2350:205d:6096:12c2:5147:a2ac	MQTT	79	Subscribe Ack (id=1)
228	08:22:36.123329	2001:41d0:1:925e::1	2402:4000:2350:205d:6096:12c2:5147:a2ac	MQTT	89	Publish Message [internet]
231	08:22:37.129003	2402:4000:2350:205d:6096:12c2:5147:a2ac	2001:41d0:1:925e::1	MQTT	76	Ping Request
232	08:22:37.456574	2001:41d0:1:925e::1	2402:4000:2350:205d:6096:12c2:5147:a2ac	MQTT	76	Ping Response

```
Frame 228: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, id 0
Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
Transmission Control Protocol, Src Port: 1883, Dst Port: 50677, Seq: 10, Ack: 39, Len: 15
MQ Telemetry Transport Protocol, Publish Message
    Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)
    Msg Len: 13
    Topic Length: 8
    Topic: internet
    Message: 494f54
```

```
0000  e8 84 a5 3e dd 57 d8 d8  66 0f 19 c9 86 dd 60 03   ···>·W··f····
0010  ce 7c 00 23 06 34 20 01  41 d0 00 01 92 5e 00 00   ·|·#·4··A····
0020  00 00 00 00 00 01 24 02  40 00 23 50 20 5d 60 96   ······$·@·#P
0030  12 c2 51 47 a2 ac 07 5b  c5 f5 06 a7 42 28 a2 0f   ··QG···[····B
0040  4e dd 50 18 01 fb 20 6d  00 00 30 0d 00 08 69 6e   N·P····m··0··
0050  74 65 72 6e 65 74 49 4f  54                        ternetIO T
```

The sequence starts with Computer B sending a Connect Command to the MQTT broker using TCP port 1883.

The connection is established, as indicated by the subsequent Connect Acknowledgment (ACK).

Computer B sends a Subscribe Request for the topic " internet " to the MQTT broker.

The broker acknowledges this subscription with a Subscribe Acknowledgment (ACK).

The MQTT broker delivers this published message to all subscribed clients, including Computer B, which has already subscribed to the internet topic. The message content is IOT, as decoded from the hexadecimal payload 494f54.

The QoS level of 0 indicates that the message is delivered at most once, with no acknowledgment or retry, which is a "Fire and Forget" delivery method.

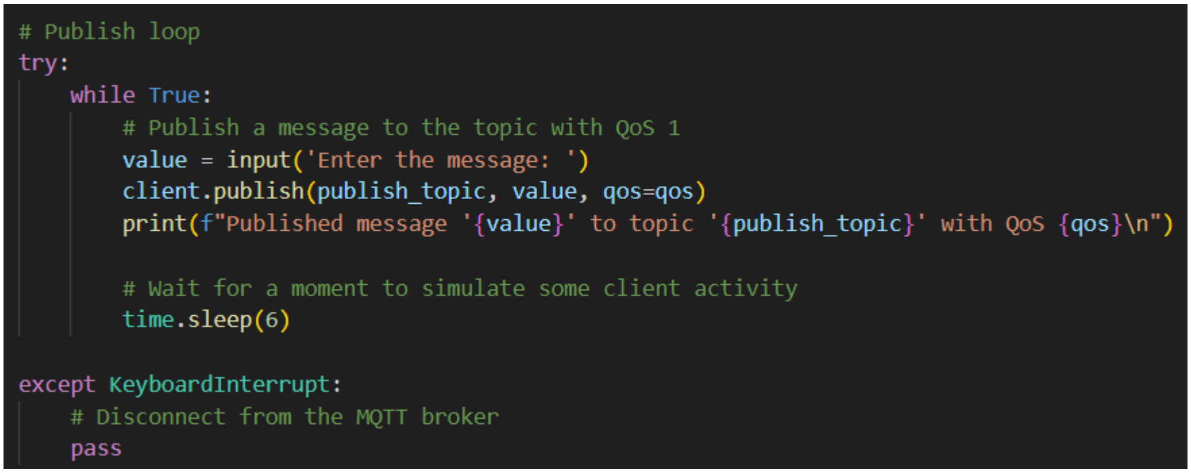
The Ping Requests and Ping Responses observed in the capture are part of the keep-alive mechanism in MQTT to ensure the connection remains open and active.

Step 3

Computer A – Publisher

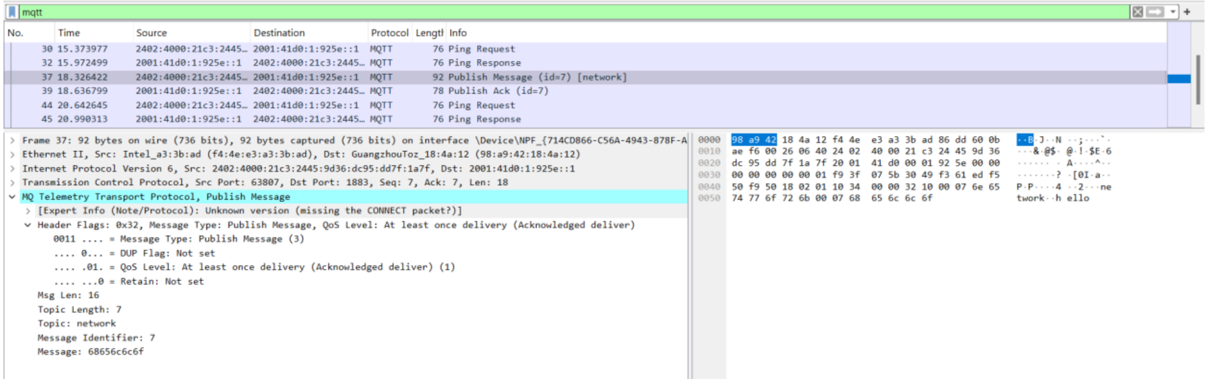
Publisher Loop was updated in the following:

- **QOS=1**



```
# Publish loop
try:
    while True:
        # Publish a message to the topic with QoS 1
        value = input('Enter the message: ')
        client.publish(publish_topic, value, qos=qos)
        print(f"Published message '{value}' to topic '{publish_topic}' with QoS {qos}\n")

        # Wait for a moment to simulate some client activity
        time.sleep(6)

except KeyboardInterrupt:
    # Disconnect from the MQTT broker
    pass
```

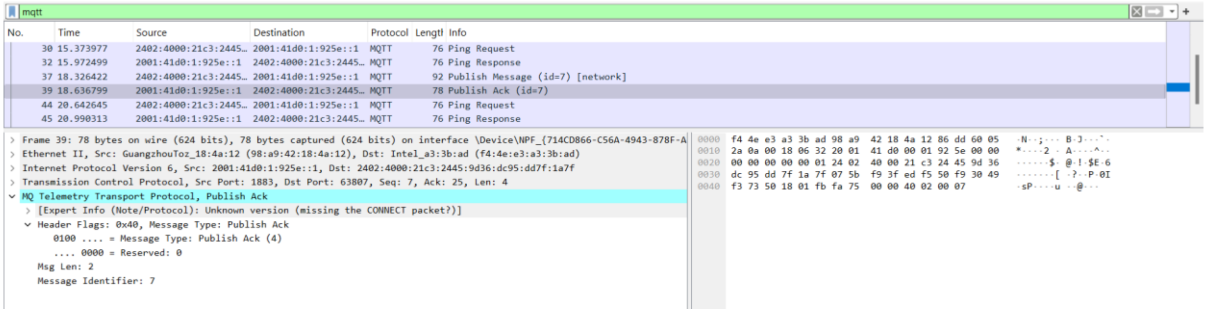
Publish message:



```
No.  Time       Source                Destination           Protocol Lengtl Info
  30 15.373977  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       76 Ping Request
  32 15.972499  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       76 Ping Response
  37 18.326422  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       92 Publish Message (id=7) [network]
  39 18.636799  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       78 Publish Ack (id=7)
  44 20.642645  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       76 Ping Request
  45 20.990313  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       76 Ping Response
```

```
> Frame 37: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface \Device\NPF_{714CD866-C56A-4943-878F-A
> Ethernet II, Src: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad), Dst: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12)
> Internet Protocol Version 6, Src: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 63807, Dst Port: 1883, Seq: 7, Ack: 7, Len: 18
v MQ Telemetry Transport Protocol, Publish Message
  > [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
  v Header Flags: 0x32, Message Type: Publish Message, QoS Level: At least once delivery (Acknowledged deliver)
      0011 .... = Message Type: Publish Message (3)
      .... 0... = DUP Flag: Not set
      .... .01. = QoS Level: At least once delivery (Acknowledged deliver) (1)
      .... ...0 = Retain: Not set
    Msg Len: 16
    Topic Length: 7
    Topic: network
    Message Identifier: 7
    Message: 68656c6c6f
```

```
0000  98 a9 42 18 4a 12 f4 4e  e3 a3 3b ad 86 dd 60 0b   ··B·J··N ··;···`·
0010  ae f6 00 26 06 40 24 02  40 00 21 c3 24 45 9d 36   ···&·@$· @·!·$E·6
0020  dc 95 dd 7f 1a 7f 20 01  41 d0 00 01 92 5e 00 00   ······ · A····^··
0030  00 00 00 00 00 01 f9 3f  07 5b 30 49 f3 61 ed f5   ·······? ·[0I·a··
0040  50 f9 50 18 02 01 10 34  00 00 32 10 00 07 6e 65   P·P····4 ··2···ne
0050  74 77 6f 72 6b 00 07 68  65 6c 6c 6f               twork··h ello
```



```
No.  Time       Source                Destination           Protocol Lengtl Info
  30 15.373977  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       76 Ping Request
  32 15.972499  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       76 Ping Response
  37 18.326422  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       92 Publish Message (id=7) [network]
  39 18.636799  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       78 Publish Ack (id=7)
  44 20.642645  2402:4000:21c3:2445…  2001:41d0:1:925e::1   MQTT       76 Ping Request
  45 20.990313  2001:41d0:1:925e::1   2402:4000:21c3:2445…  MQTT       76 Ping Response
```

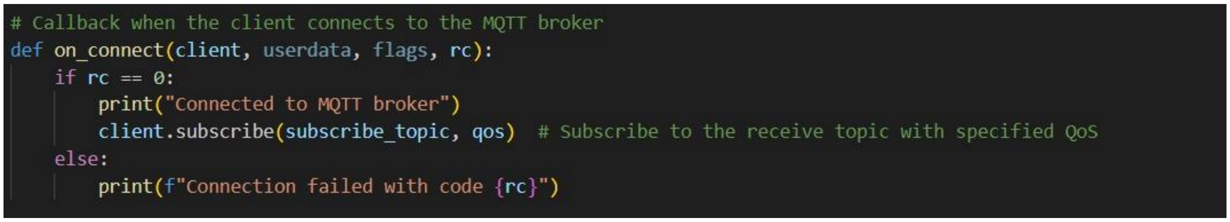
```
> Frame 39: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{714CD866-C56A-4943-878F-A
> Ethernet II, Src: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12), Dst: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f
> Transmission Control Protocol, Src Port: 1883, Dst Port: 63807, Seq: 7, Ack: 25, Len: 4
v MQ Telemetry Transport Protocol, Publish Ack
  > [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
  v Header Flags: 0x40, Message Type: Publish Ack
      0100 .... = Message Type: Publish Ack (4)
      .... 0000 = Reserved: 0
    Msg Len: 2
    Message Identifier: 7
```

```
0000  f4 4e e3 a3 3b ad 98 a9  42 18 4a 12 86 dd 60 05   ·N··;··· B·J···`·
0010  2a 0a 00 18 06 32 20 01  41 d0 00 01 92 5e 00 00   *····2 · A····^··
0020  00 00 00 00 00 01 24 02  40 00 21 c3 24 45 9d 36   ······$· @·!·$E·6
0030  dc 95 dd 7f 1a 7f 07 5b  f9 3f ed f5 50 f9 30 49   ·······[ ·?··P·0I
0040  f3 73 50 18 01 fb fa 75  00 00 40 02 00 07         ·sP····u ··@···
```

The publisher sends a PUBLISH packet to the MQTT broker with a unique message ID. The broker forwards this packet to the subscribers of the "network" topic. At QoS level 1, the publisher ensures the message is delivered at least once and requires an acknowledgment from the subscribers. If the acknowledgment isn't received, the broker will retransmit the message. In contrast, QoS level 0 offers no delivery guarantees or retries.

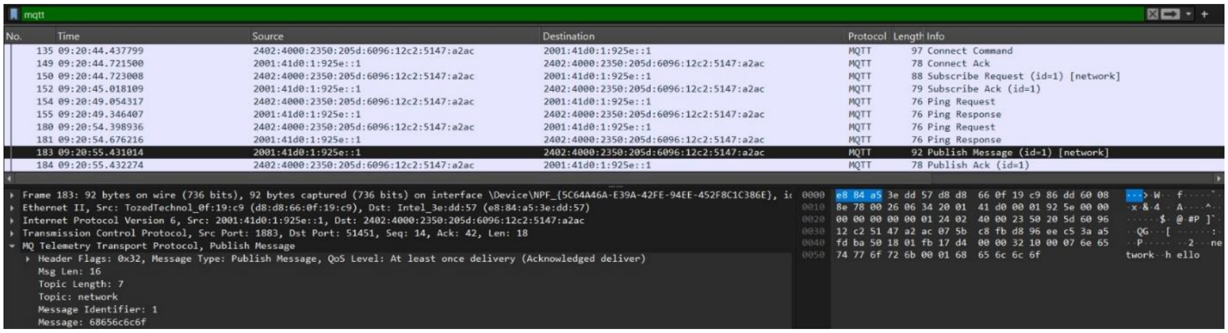
Computer B – Subscriber

The following function was updated with QOS=1



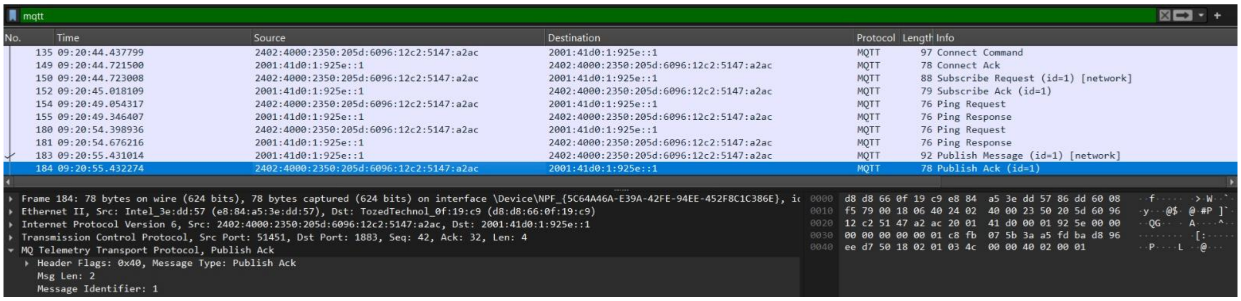
```python
# Callback when the client connects to the MQTT broker
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT broker")
        client.subscribe(subscribe_topic, qos)  # Subscribe to the receive topic with specified QoS
    else:
        print(f"Connection failed with code {rc}")
```

Received the message: Hello



```
No.  Time             Source                                  Destination                             Protocol Length Info
 135 09:20:44.437799  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     97 Connect Command
 149 09:20:44.721500  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     78 Connect Ack
 150 09:20:44.723008  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     88 Subscribe Request (id=1) [network]
 152 09:20:45.018109  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     79 Subscribe Ack (id=1)
 154 09:20:49.054317  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     76 Ping Request
 155 09:20:49.346407  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     76 Ping Response
 180 09:20:54.398936  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     76 Ping Request
 181 09:20:54.676216  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     76 Ping Response
 183 09:20:55.431014  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     92 Publish Message (id=1) [network]
 184 09:20:55.432274  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     78 Publish Ack (id=1)

▸ Frame 183: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, i
▸ Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
▸ Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
▸ Transmission Control Protocol, Src Port: 1883, Dst Port: 51451, Seq: 14, Ack: 42, Len: 18
▾ MQ Telemetry Transport Protocol, Publish Message
  ▸ Header Flags: 0x32, Message Type: Publish Message, QoS Level: At least once delivery (Acknowledged deliver)
    Msg Len: 16
    Topic Length: 7
    Topic: network
    Message Identifier: 1
    Message: 68656c6c6f

0000  e8 84 a5 3e dd 57 d8 d8  66 0f 19 c9 86 dd 60 08   ···>·W·· f·····`·
0010  8e 78 00 26 06 34 20 01  41 d0 00 01 92 5e 00 00   ·x·&·4 · A····^··
0020  00 00 00 00 00 01 24 02  40 00 23 50 20 5d 60 96   ······$· @·#P ]`·
0030  12 c2 51 47 a2 ac 07 5b  c8 fb d8 96 ee c5 3a a5   ··QG···[ ······:·
0040  fd ba 50 18 01 fb 17 d4  00 00 32 10 00 07 6e 65   ··P····· ··2···ne
0050  74 77 6f 72 6b 00 01 68  65 6c 6c 6f               twork··h ello
```

Acknowledgment:



```
No.  Time             Source                                  Destination                             Protocol Length Info
 135 09:20:44.437799  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     97 Connect Command
 149 09:20:44.721500  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     78 Connect Ack
 150 09:20:44.723008  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     88 Subscribe Request (id=1) [network]
 152 09:20:45.018109  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     79 Subscribe Ack (id=1)
 154 09:20:49.054317  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     76 Ping Request
 155 09:20:49.346407  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     76 Ping Response
 180 09:20:54.398936  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     76 Ping Request
 181 09:20:54.676216  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     76 Ping Response
 183 09:20:55.431014  2001:41d0:1:925e::1                     2402:4000:2350:205d:6096:12c2:5147:a2ac MQTT     92 Publish Message (id=1) [network]
 184 09:20:55.432274  2402:4000:2350:205d:6096:12c2:5147:a2ac 2001:41d0:1:925e::1                     MQTT     78 Publish Ack (id=1)

▸ Frame 184: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{5C64A46A-E39A-42FE-94EE-452F8C1C386E}, i
▸ Ethernet II, Src: Intel_3e:dd:57 (e8:84:a5:3e:dd:57), Dst: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9)
▸ Internet Protocol Version 6, Src: 2402:4000:2350:205d:6096:12c2:5147:a2ac, Dst: 2001:41d0:1:925e::1
▸ Transmission Control Protocol, Src Port: 51451, Dst Port: 1883, Seq: 42, Ack: 32, Len: 4
▾ MQ Telemetry Transport Protocol, Publish Ack
  ▸ Header Flags: 0x40, Message Type: Publish Ack
    Msg Len: 2
    Message Identifier: 1

0000  d8 d8 66 0f 19 c9 e8 84  a5 3e dd 57 86 dd 60 08   ··f····· ·>·W··`·
0010  f5 79 00 18 06 40 24 02  40 00 23 50 20 5d 60 96   ·y···@$· @·#P ]`·
0020  12 c2 51 47 a2 ac 20 01  41 d0 00 01 92 5e 00 00   ··QG·· · A····^··
0030  00 00 00 00 00 01 c8 fb  07 5b 3a a5 fd ba d8 96   ········ ·[:·····
0040  ee d7 50 18 02 01 03 4c  00 00 40 02 00 01         ··P····L ··@···
```

The broker forwards the PUBLISH packet to the subscriber who is subscribed to the relevant topic "network". A unique ID assigned by the broker, used to track the message and its acknowledgment. In this case the ID assigned by broker is 1.

The subscriber sends a PUBACK packet back to the broker to acknowledge receipt of the message.

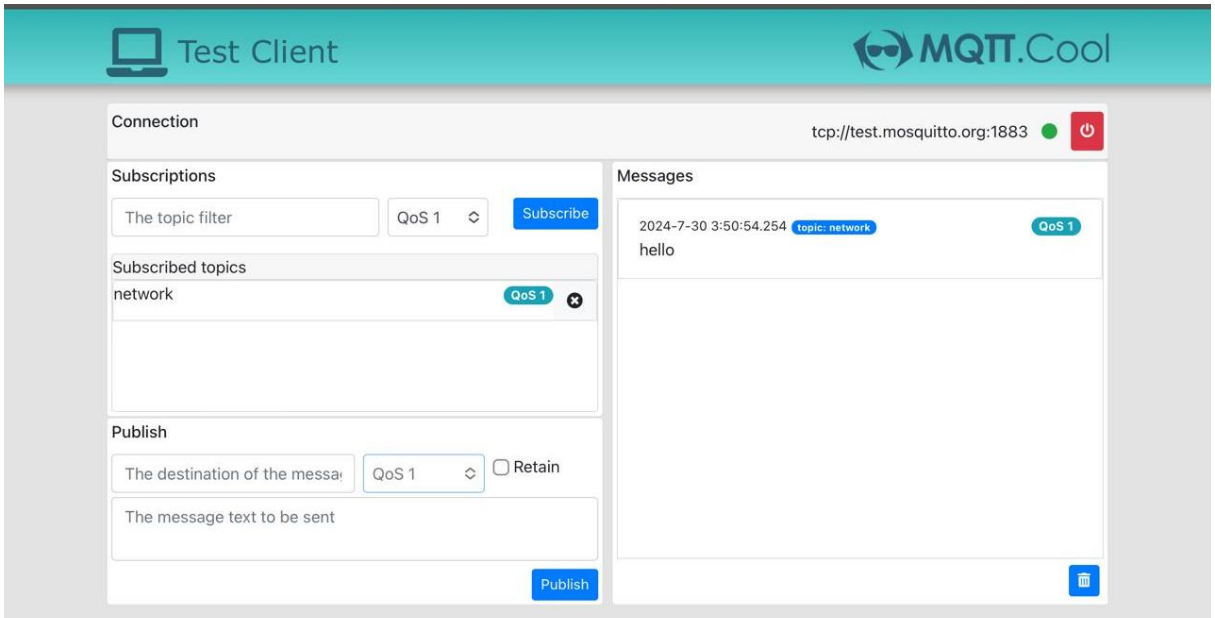
The QoS level 1 ensures that the message is delivered at least once, with confirmation required (Acknowledged delivery). In contrast, QoS level 0 means the message is delivered at most once, with no guarantee of delivery.

QoS 1: Ensures that the message is delivered at least once, with an acknowledgment required from the receiver. This provides reliable delivery but may result in duplicate messages if the acknowledgment is delayed or lost.

QoS 0: Delivers the message at most once without requiring any acknowledgment, making it fast and efficient but with no guarantee of delivery.

Computer C – Broker

Broker was subscribed to topic "network" with QoS 1



Connection: tcp://test.mosquitto.org:1883 ●

Subscriptions: QoS 1

Subscribed topics: network QoS 1

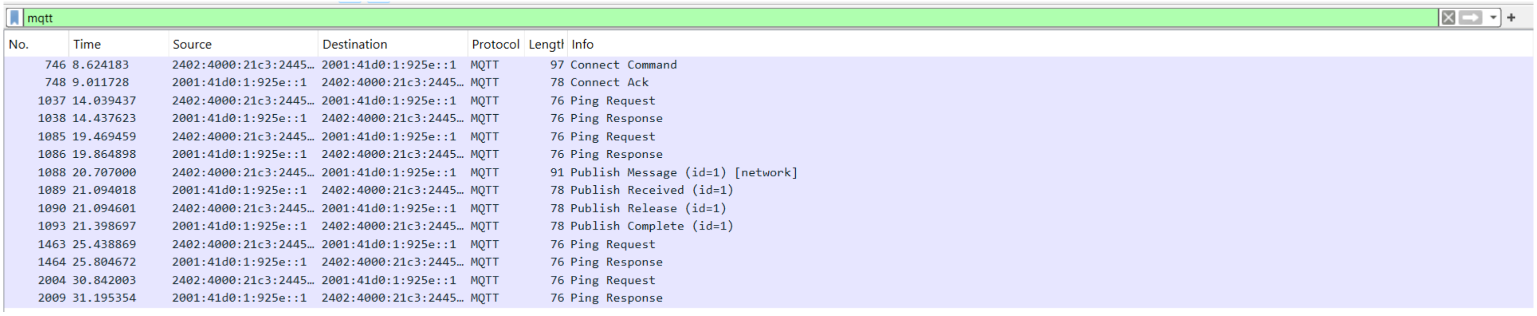
Publish: QoS 1

Messages:

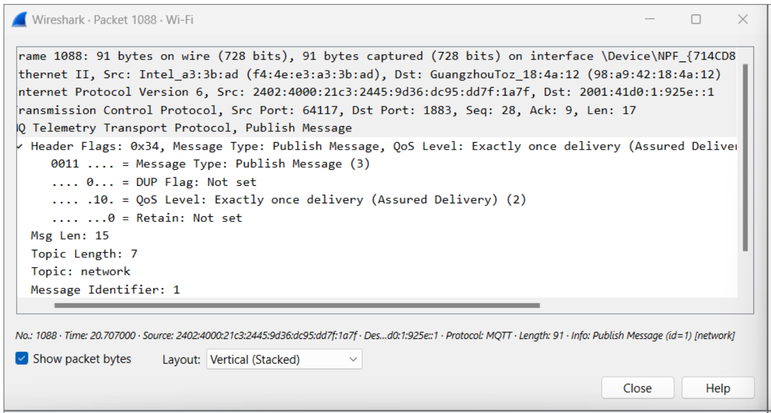
2024-7-30 3:50:54.254 topic: network QoS 1
hello

- **<u>QOS=2</u>**

Computer A – Publisher



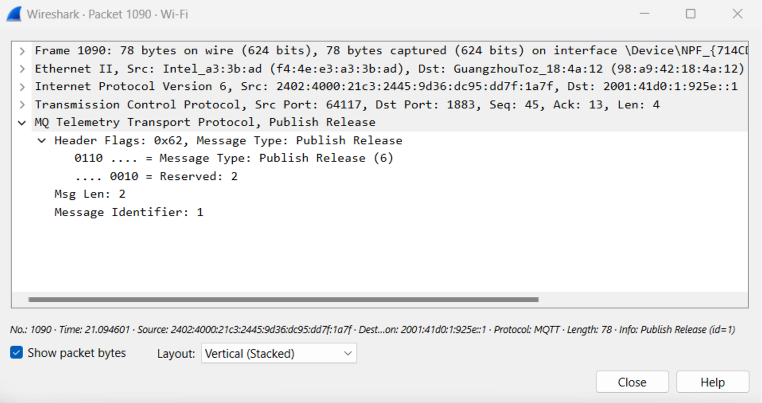
```
No.    Time       Source                Destination           Protocol Length Info
   746 8.624183   2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       97 Connect Command
   748 9.011728   2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       78 Connect Ack
  1037 14.039437  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       76 Ping Request
  1038 14.437623  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       76 Ping Response
  1085 19.469459  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       76 Ping Request
  1086 19.864898  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       76 Ping Response
  1088 20.707000  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       91 Publish Message (id=1) [network]
  1089 21.094018  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       78 Publish Received (id=1)
  1090 21.094601  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       78 Publish Release (id=1)
  1093 21.398697  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       78 Publish Complete (id=1)
  1463 25.438869  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       76 Ping Request
  1464 25.804672  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       76 Ping Response
  2004 30.842003  2402:4000:21c3:2445… 2001:41d0:1:925e::1   MQTT       76 Ping Request
  2009 31.195354  2001:41d0:1:925e::1   2402:4000:21c3:2445… MQTT       76 Ping Response
```



```
rame 1088: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface \Device\NPF_{714CD8
thernet II, Src: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad), Dst: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12)
nternet Protocol Version 6, Src: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f, Dst: 2001:41d0:1:925e::1
ransmission Control Protocol, Src Port: 64117, Dst Port: 1883, Seq: 28, Ack: 9, Len: 17
Q Telemetry Transport Protocol, Publish Message
 Header Flags: 0x34, Message Type: Publish Message, QoS Level: Exactly once delivery (Assured Delive
    0011 .... = Message Type: Publish Message (3)
    .... 0... = DUP Flag: Not set
    .... .10. = QoS Level: Exactly once delivery (Assured Delivery) (2)
    .... ...0 = Retain: Not set
  Msg Len: 15
  Topic Length: 7
  Topic: network
  Message Identifier: 1
```

No.: 1088 · Time: 20.707000 · Source: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f · Des...d0:1:925e::1 · Protocol: MQTT · Length: 91 · Info: Publish Message (id=1) [network]

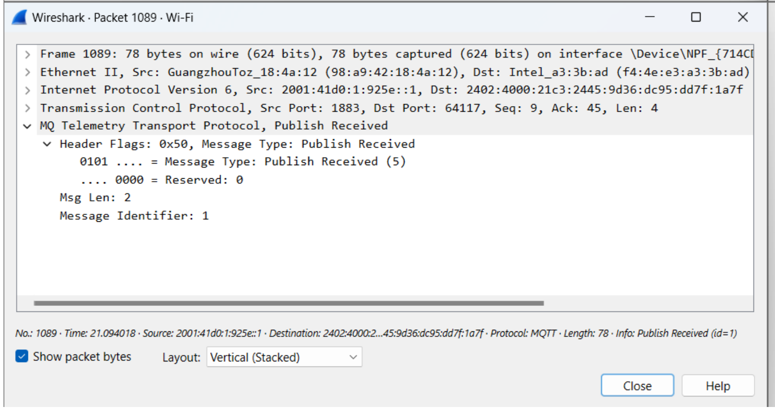
☑ Show packet bytes   Layout: Vertical (Stacked)



```
> Frame 1090: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{714CI
> Ethernet II, Src: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad), Dst: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12)
> Internet Protocol Version 6, Src: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f, Dst: 2001:41d0:1:925e::1
> Transmission Control Protocol, Src Port: 64117, Dst Port: 1883, Seq: 45, Ack: 13, Len: 4
v MQ Telemetry Transport Protocol, Publish Release
  v Header Flags: 0x62, Message Type: Publish Release
       0110 .... = Message Type: Publish Release (6)
       .... 0010 = Reserved: 2
    Msg Len: 2
    Message Identifier: 1
```

No.: 1090 · Time: 21.094601 · Source: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f · Dest...on: 2001:41d0:1:925e::1 · Protocol: MQTT · Length: 78 · Info: Publish Release (id=1)

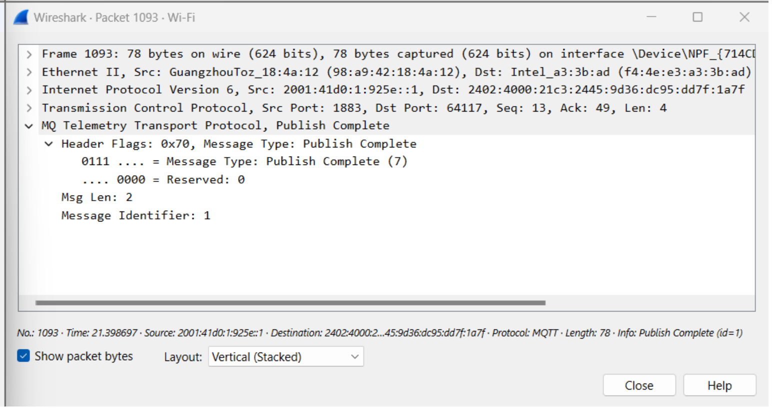
☑ Show packet bytes   Layout: Vertical (Stacked)



```
> Frame 1089: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{714CI
> Ethernet II, Src: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12), Dst: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f
> Transmission Control Protocol, Src Port: 1883, Dst Port: 64117, Seq: 9, Ack: 45, Len: 4
v MQ Telemetry Transport Protocol, Publish Received
  v Header Flags: 0x50, Message Type: Publish Received
       0101 .... = Message Type: Publish Received (5)
       .... 0000 = Reserved: 0
    Msg Len: 2
    Message Identifier: 1
```

No.: 1089 · Time: 21.094018 · Source: 2001:41d0:1:925e::1 · Destination: 2402:4000:2...45:9d36:dc95:dd7f:1a7f · Protocol: MQTT · Length: 78 · Info: Publish Received (id=1)

☑ Show packet bytes   Layout: Vertical (Stacked)

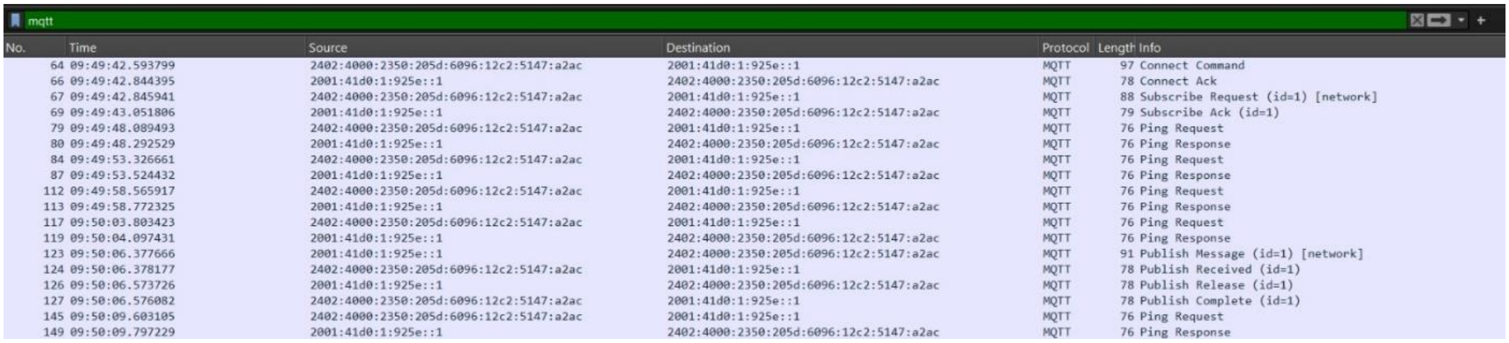


```
> Frame 1093: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{714CI
> Ethernet II, Src: GuangzhouToz_18:4a:12 (98:a9:42:18:4a:12), Dst: Intel_a3:3b:ad (f4:4e:e3:a3:3b:ad)
> Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:21c3:2445:9d36:dc95:dd7f:1a7f
> Transmission Control Protocol, Src Port: 1883, Dst Port: 64117, Seq: 13, Ack: 49, Len: 4
v MQ Telemetry Transport Protocol, Publish Complete
  v Header Flags: 0x70, Message Type: Publish Complete
       0111 .... = Message Type: Publish Complete (7)
       .... 0000 = Reserved: 0
    Msg Len: 2
    Message Identifier: 1
```

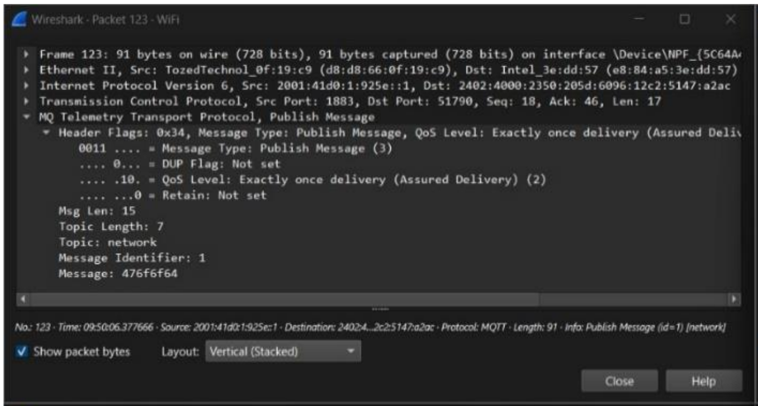
No.: 1093 · Time: 21.398697 · Source: 2001:41d0:1:925e::1 · Destination: 2402:4000:2...45:9d36:dc95:dd7f:1a7f · Protocol: MQTT · Length: 78 · Info: Publish Complete (id=1)

☑ Show packet bytes   Layout: Vertical (Stacked)

Computer B – Subscriber



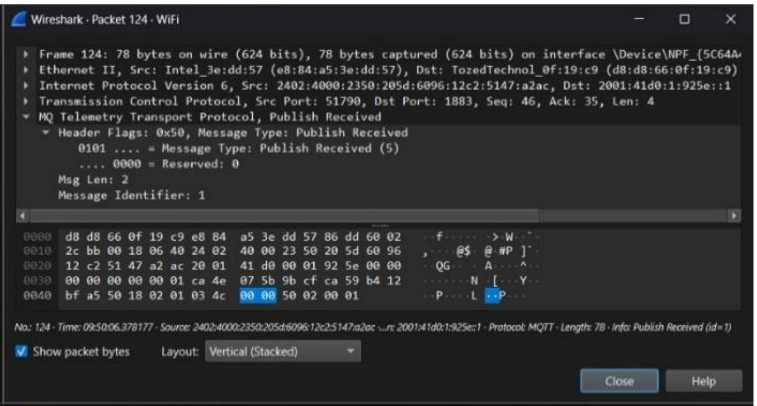
```
No.   Time             Source                                   Destination                              Protocol Length Info
   64 09:49:42.593799  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     97 Connect Command
   66 09:49:42.844395  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Connect Ack
   67 09:49:42.845941  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     88 Subscribe Request (id=1) [network]
   69 09:49:43.051806  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     79 Subscribe Ack (id=1)
   79 09:49:48.089493  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
   80 09:49:48.292529  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response
   84 09:49:53.326661  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
   87 09:49:53.524432  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response
  112 09:49:58.565917  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
  113 09:49:58.772325  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response
  117 09:50:03.803423  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
  119 09:50:04.097431  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response
  123 09:50:06.377666  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     91 Publish Message (id=1) [network]
  124 09:50:06.378177  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     78 Publish Received (id=1)
  126 09:50:06.573726  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     78 Publish Release (id=1)
  127 09:50:06.576082  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     78 Publish Complete (id=1)
  145 09:50:09.603105  2402:4000:2350:205d:6096:12c2:5147:a2ac  2001:41d0:1:925e::1                      MQTT     76 Ping Request
  149 09:50:09.797229  2001:41d0:1:925e::1                      2402:4000:2350:205d:6096:12c2:5147:a2ac  MQTT     76 Ping Response
```



```
▸ Frame 123: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface \Device\NPF_{5C64A
▸ Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
▸ Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
▸ Transmission Control Protocol, Src Port: 1883, Dst Port: 51790, Seq: 18, Ack: 46, Len: 17
▾ MQ Telemetry Transport Protocol, Publish Message
  ▾ Header Flags: 0x34, Message Type: Publish Message, QoS Level: Exactly once delivery (Assured Deliv
      0011 .... = Message Type: Publish Message (3)
      .... 0... = DUP Flag: Not set
      .... .10. = QoS Level: Exactly once delivery (Assured Delivery) (2)
      .... ...0 = Retain: Not set
    Msg Len: 15
    Topic Length: 7
    Topic: network
    Message Identifier: 1
    Message: 476f6f64
```

No.: 123 · Time: 09:50:06.377666 · Source: 2001:41d0:1:925e::1 · Destination: 2402:4...2c2:5147:a2ac · Protocol: MQTT · Length: 91 · Info: Publish Message (id=1) [network]

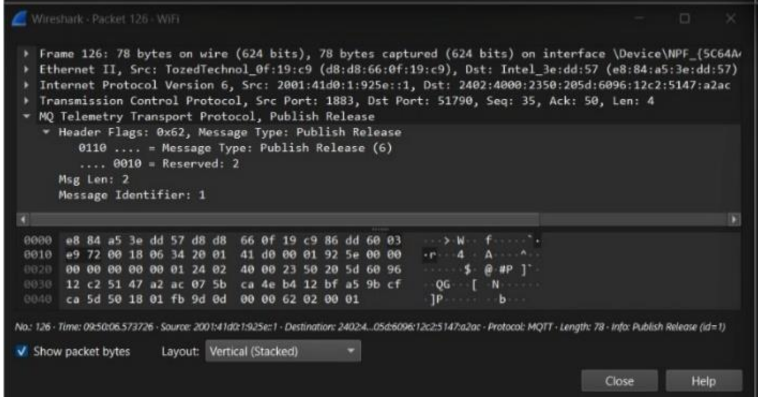
☑ Show packet bytes   Layout: Vertical (Stacked)



```
▸ Frame 124: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{5C64A
▸ Ethernet II, Src: Intel_3e:dd:57 (e8:84:a5:3e:dd:57), Dst: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9)
▸ Internet Protocol Version 6, Src: 2402:4000:2350:205d:6096:12c2:5147:a2ac, Dst: 2001:41d0:1:925e::1
▸ Transmission Control Protocol, Src Port: 51790, Dst Port: 1883, Seq: 46, Ack: 35, Len: 4
▾ MQ Telemetry Transport Protocol, Publish Received
  ▾ Header Flags: 0x50, Message Type: Publish Received
      0101 .... = Message Type: Publish Received (5)
      .... 0000 = Reserved: 0
    Msg Len: 2
    Message Identifier: 1

0000  d8 d8 66 0f 19 c9 e8 84  a5 3e dd 57 86 dd 60 02   ··f·····>·W··`·
0010  2c bb 00 18 06 40 24 02  40 00 23 50 20 5d 60 96   ,····@$·@·#P ]`·
0020  12 c2 51 47 a2 ac 20 01  41 d0 00 01 92 5e 00 00   ··QG·· ·A····^··
0030  00 00 00 00 00 01 ca 4e  07 5b 9b cf ca 59 b4 12   ·······N·[···Y··
0040  bf a5 50 18 02 01 03 4c  00 00 50 02 00 01         ··P····L··P···
```

No.: 124 · Time: 09:50:06.378177 · Source: 2402:4000:2350:205d:6096:12c2:5147:a2ac ·...n: 2001:41d0:1:925e::1 · Protocol: MQTT · Length: 78 · Info: Publish Received (id=1)

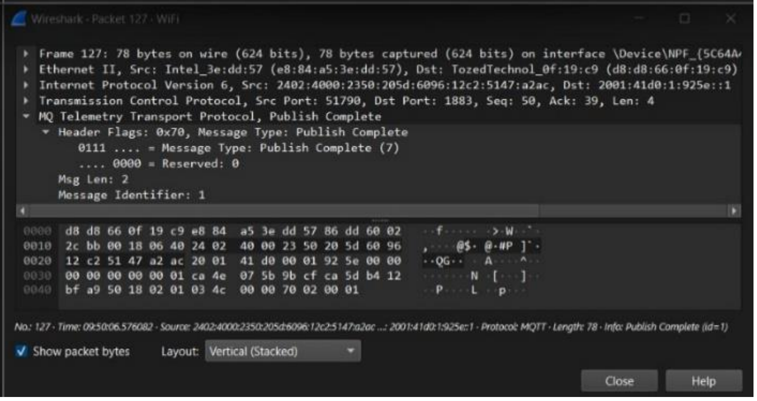
☑ Show packet bytes   Layout: Vertical (Stacked)



```
▸ Frame 126: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{5C64A
▸ Ethernet II, Src: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9), Dst: Intel_3e:dd:57 (e8:84:a5:3e:dd:57)
▸ Internet Protocol Version 6, Src: 2001:41d0:1:925e::1, Dst: 2402:4000:2350:205d:6096:12c2:5147:a2ac
▸ Transmission Control Protocol, Src Port: 1883, Dst Port: 51790, Seq: 35, Ack: 50, Len: 4
▾ MQ Telemetry Transport Protocol, Publish Release
  ▾ Header Flags: 0x62, Message Type: Publish Release
      0110 .... = Message Type: Publish Release (6)
      .... 0010 = Reserved: 2
    Msg Len: 2
    Message Identifier: 1

0000  e8 84 a5 3e dd 57 d8 d8  66 0f 19 c9 86 dd 60 03   ···>·W··f·····`·
0010  e9 72 00 18 06 34 20 01  41 d0 00 01 92 5e 00 00   ·r···4 ·A····^··
0020  00 00 00 00 00 01 24 02  40 00 23 50 20 5d 60 96   ······$·@·#P ]`·
0030  12 c2 51 47 a2 ac 07 5b  ca 4e b4 12 bf a5 9b cf   ··QG···[·N······
0040  ca 5d 50 18 01 fb 9d 0d  00 00 62 02 00 01         ·]P·······b···
```

No.: 126 · Time: 09:50:06.573726 · Source: 2001:41d0:1:925e::1 · Destination: 2402:4...05d:6096:12c2:5147:a2ac · Protocol: MQTT · Length: 78 · Info: Publish Release (id=1)

☑ Show packet bytes   Layout: Vertical (Stacked)

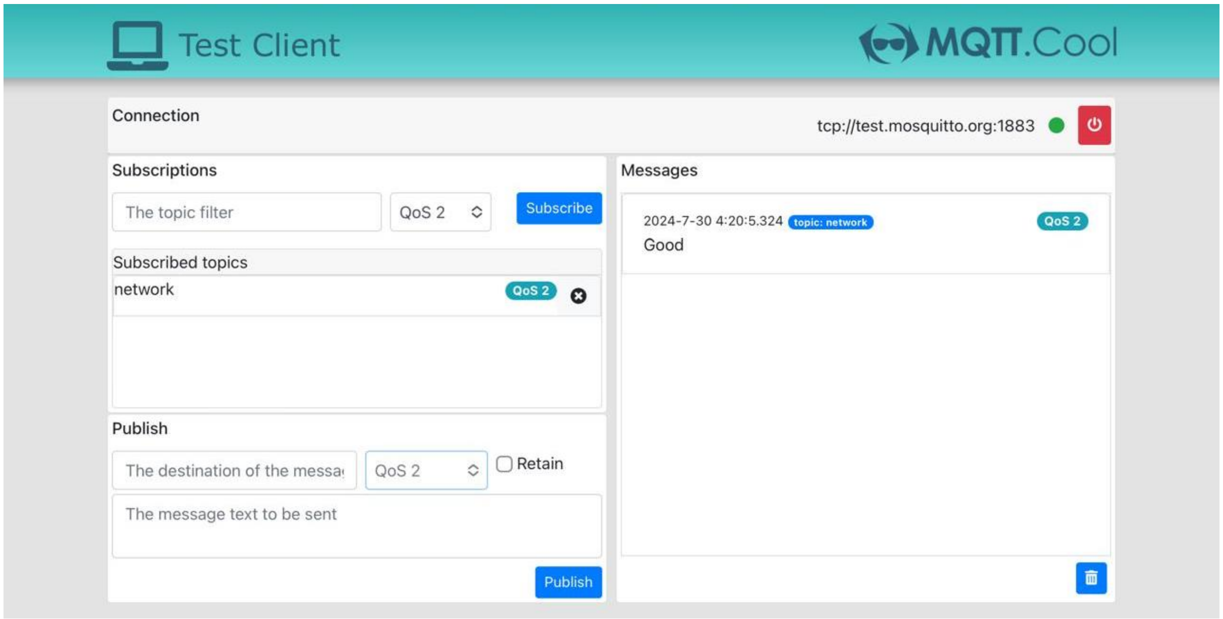


```
▸ Frame 127: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface \Device\NPF_{5C64A
▸ Ethernet II, Src: Intel_3e:dd:57 (e8:84:a5:3e:dd:57), Dst: TozedTechnol_0f:19:c9 (d8:d8:66:0f:19:c9)
▸ Internet Protocol Version 6, Src: 2402:4000:2350:205d:6096:12c2:5147:a2ac, Dst: 2001:41d0:1:925e::1
▸ Transmission Control Protocol, Src Port: 51790, Dst Port: 1883, Seq: 50, Ack: 39, Len: 4
▾ MQ Telemetry Transport Protocol, Publish Complete
  ▾ Header Flags: 0x70, Message Type: Publish Complete
      0111 .... = Message Type: Publish Complete (7)
      .... 0000 = Reserved: 0
    Msg Len: 2
    Message Identifier: 1

0000  d8 d8 66 0f 19 c9 e8 84  a5 3e dd 57 86 dd 60 02   ··f·····>·W··`·
0010  2c bb 00 18 06 40 24 02  40 00 23 50 20 5d 60 96   ,····@$·@·#P ]`·
0020  12 c2 51 47 a2 ac 20 01  41 d0 00 01 92 5e 00 00   ··QG·· ·A····^··
0030  00 00 00 00 00 01 ca 4e  07 5b 9b cf ca 5d b4 12   ·······N·[···]··
0040  bf a9 50 18 02 01 03 4c  00 00 70 02 00 01         ··P····L··p···
```

No.: 127 · Time: 09:50:06.576082 · Source: 2402:4000:2350:205d:6096:12c2:5147:a2ac ...: 2001:41d0:1:925e::1 · Protocol: MQTT · Length: 78 · Info: Publish Complete (id=1)

☑ Show packet bytes   Layout: Vertical (Stacked)

Computer C - Broker



tcp://test.mosquitto.org:1883

Subscriptions: QoS 2

Subscribed topics: network QoS 2

Publish: QoS 2

Messages:

2024-7-30 4:20:5.324 topic: network QoS 2
Good

The publisher sends a PUBLISH packet with the QoS level set to 2, indicating that the message should be delivered exactly once.

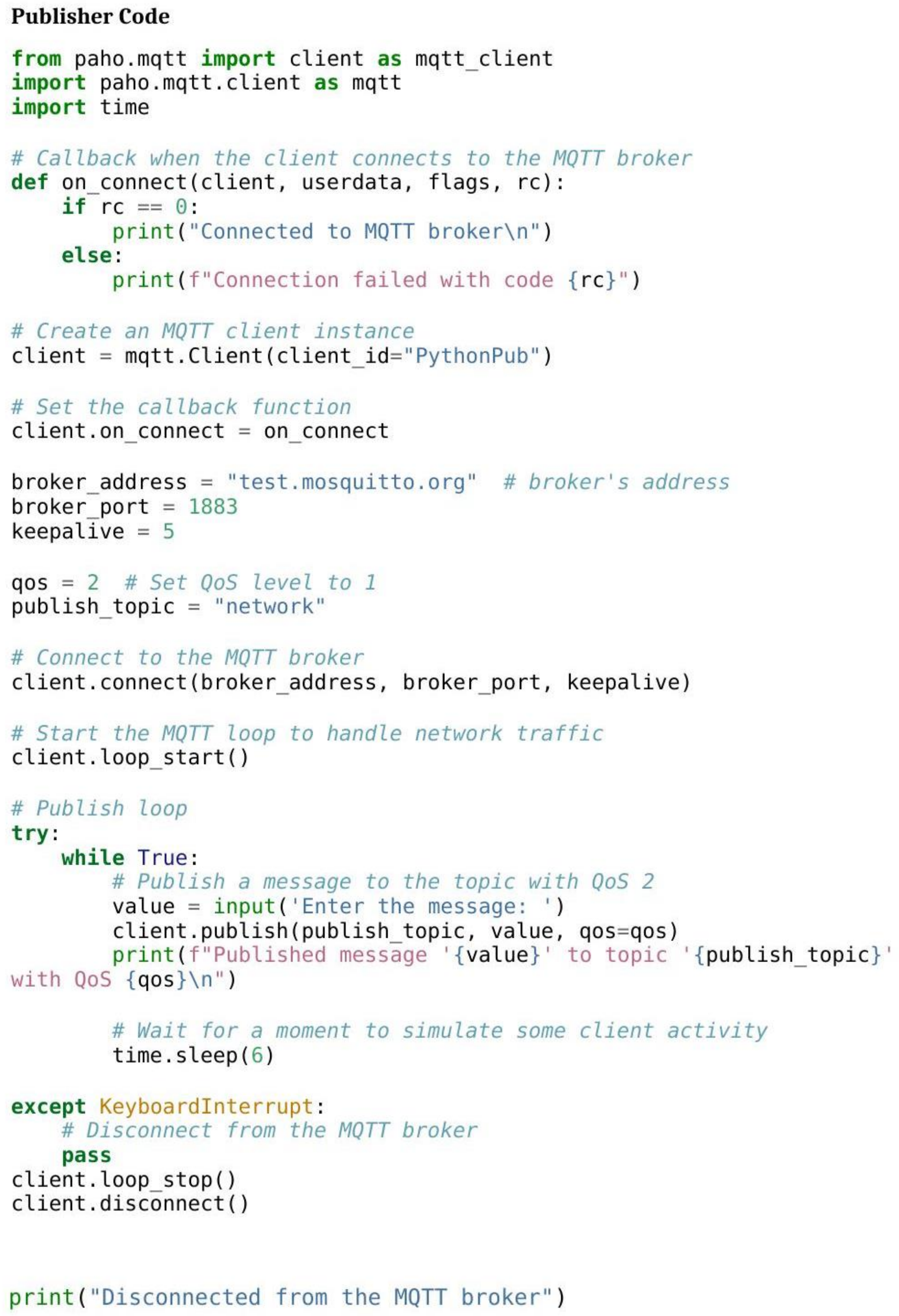
The QoS level of 2 indicates that the message is exactly once delivery (Assured delivery).

The subscriber (or broker, if intermediate) responds with a PUBREC (Publish Received) packet to acknowledge that it has received the message but has not yet completed processing it.

The publisher (or broker) responds to the PUBREC with a PUBREL (Publish Release) packet, indicating that it is ready to proceed to the next step.

The subscriber (or broker) sends a PUBCOMP (Publish Complete) packet, confirming that the message has been fully received and processed.

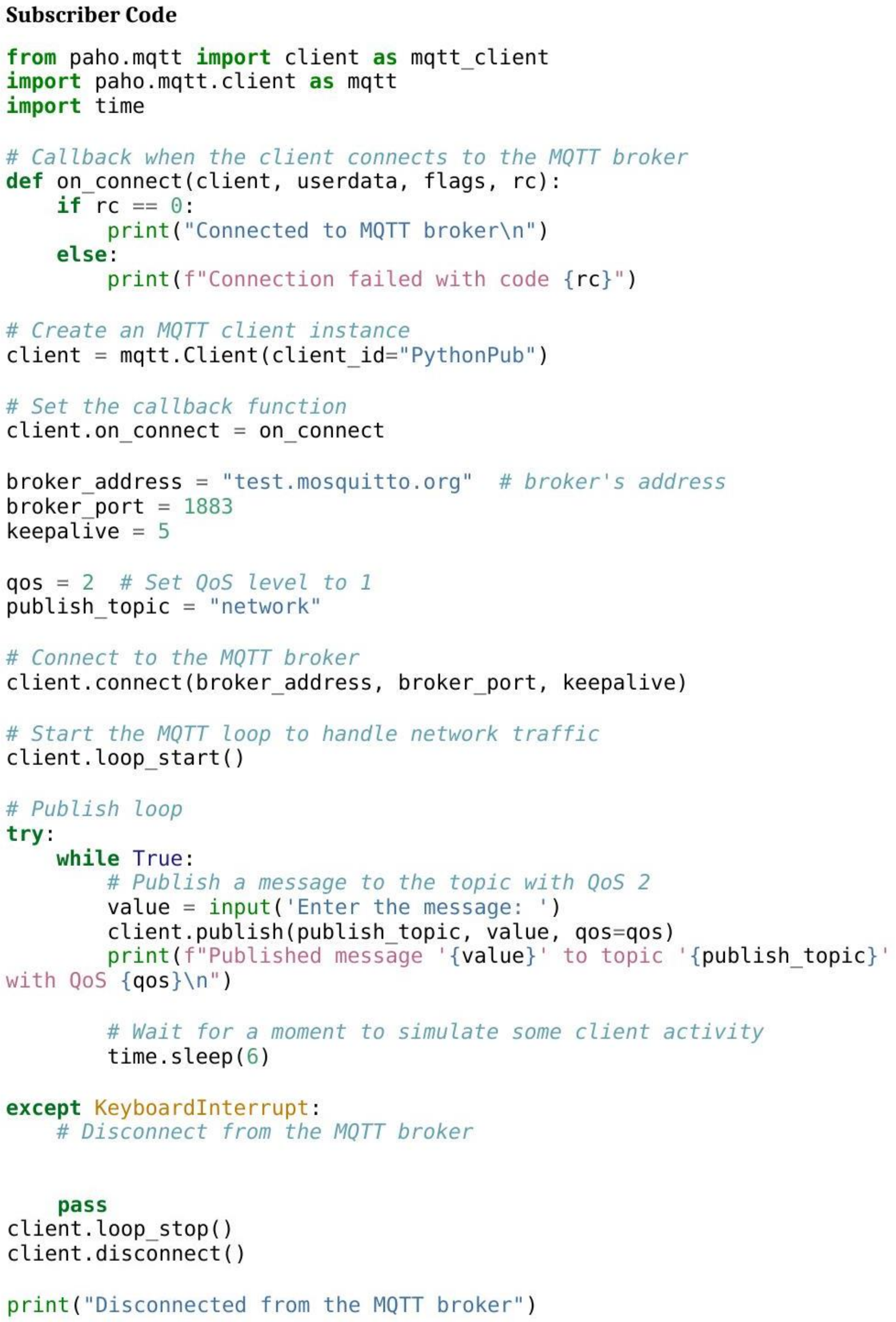
Publisher Code



Publisher Code

```
from paho.mqtt import client as mqtt_client
import paho.mqtt.client as mqtt
import time

# Callback when the client connects to the MQTT broker
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT broker\n")
    else:
        print(f"Connection failed with code {rc}")

# Create an MQTT client instance
client = mqtt.Client(client_id="PythonPub")

# Set the callback function
client.on_connect = on_connect

broker_address = "test.mosquitto.org"  # broker's address
broker_port = 1883
keepalive = 5

qos = 2  # Set QoS level to 1
publish_topic = "network"

# Connect to the MQTT broker
client.connect(broker_address, broker_port, keepalive)

# Start the MQTT loop to handle network traffic
client.loop_start()

# Publish loop
try:
    while True:
        # Publish a message to the topic with QoS 2
        value = input('Enter the message: ')
        client.publish(publish_topic, value, qos=qos)
        print(f"Published message '{value}' to topic '{publish_topic}'
with QoS {qos}\n")

        # Wait for a moment to simulate some client activity
        time.sleep(6)

except KeyboardInterrupt:
    # Disconnect from the MQTT broker
    pass
client.loop_stop()
client.disconnect()



print("Disconnected from the MQTT broker")
```

Publisher Code



Subscriber Code

```
from paho.mqtt import client as mqtt_client
import paho.mqtt.client as mqtt
import time

# Callback when the client connects to the MQTT broker
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT broker\n")
    else:
        print(f"Connection failed with code {rc}")

# Create an MQTT client instance
client = mqtt.Client(client_id="PythonPub")

# Set the callback function
client.on_connect = on_connect

broker_address = "test.mosquitto.org"  # broker's address
broker_port = 1883
keepalive = 5

qos = 2  # Set QoS level to 1
publish_topic = "network"

# Connect to the MQTT broker
client.connect(broker_address, broker_port, keepalive)

# Start the MQTT loop to handle network traffic
client.loop_start()

# Publish loop
try:
    while True:
        # Publish a message to the topic with QoS 2
        value = input('Enter the message: ')
        client.publish(publish_topic, value, qos=qos)
        print(f"Published message '{value}' to topic '{publish_topic}'
with QoS {qos}\n")

        # Wait for a moment to simulate some client activity
        time.sleep(6)

except KeyboardInterrupt:
    # Disconnect from the MQTT broker


    pass
client.loop_stop()
client.disconnect()

print("Disconnected from the MQTT broker")
```

Homework

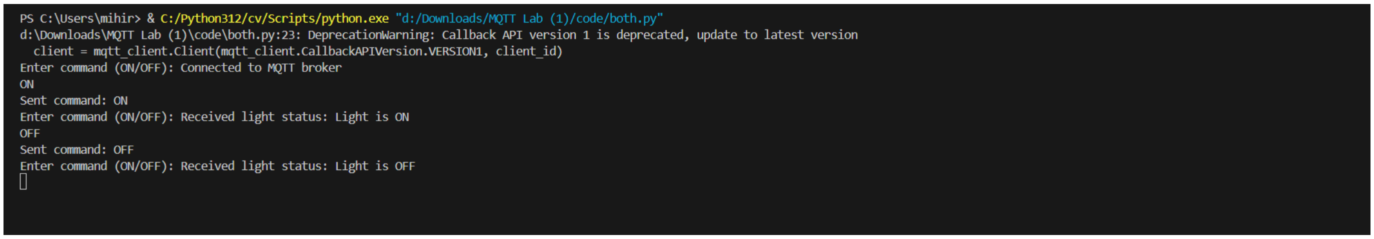
MQTT-Based Light Control and Monitoring System

This application demonstrates a basic implementation of an MQTT-based system for controlling and monitoring the status of a light. It consists of two main components:

Central Dashboard: Acts as the user interface for sending commands to control the light and receiving status updates. This component enables the user to interact with the system by issuing commands like "ON" or "OFF" and viewing the current status of the light.

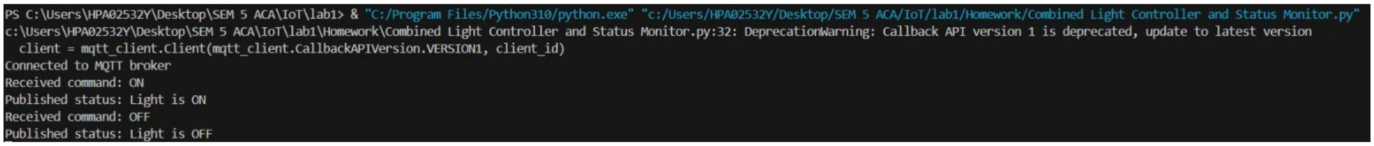
Light Controller and Status Monitor Component: Manages the actual light control and monitors its status. It receives commands from the Central Dashboard, processes them to control the light's state, and publishes status updates back to the Central Dashboard.

- Central Dashboard



```
PS C:\Users\mihir> & C:/Python312/cv/Scripts/python.exe "d:/Downloads/MQTT Lab (1)/code/both.py"
d:\Downloads\MQTT Lab (1)\code\both.py:23: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt_client.Client(mqtt_client.CallbackAPIVersion.VERSION1, client_id)
Enter command (ON/OFF): Connected to MQTT broker
ON
Sent command: ON
Enter command (ON/OFF): Received light status: Light is ON
OFF
Sent command: OFF
Enter command (ON/OFF): Received light status: Light is OFF
```

- Light Controller and Status Monitor Component:



```
PS C:\Users\HPA02532Y\Desktop\SEM 5 ACA\IoT\lab1> & "C:/Program Files/Python310/python.exe" "c:/Users/HPA02532Y/Desktop/SEM 5 ACA/IoT/lab1/Homework/Combined Light Controller and Status Monitor.py"
c:\Users\HPA02532Y\Desktop\SEM 5 ACA\IoT\lab1\Homework\Combined Light Controller and Status Monitor.py:32: DeprecationWarning: Callback API version 1 is deprecated, update to latest version
  client = mqtt_client.Client(mqtt_client.CallbackAPIVersion.VERSION1, client_id)
Connected to MQTT broker
Received command: ON
Published status: Light is ON
Received command: OFF
Published status: Light is OFF
```

Workflow

Turning On the Light:

The Central Hub sends a "TURN ON" command to the Light Controller via the MQTT broker.

The Light Controller receives this command, turns on the light bulb, and then publishes a status update indicating that the light is now ON.

The Central Hub receives this status update and displays the light's current state as ON.

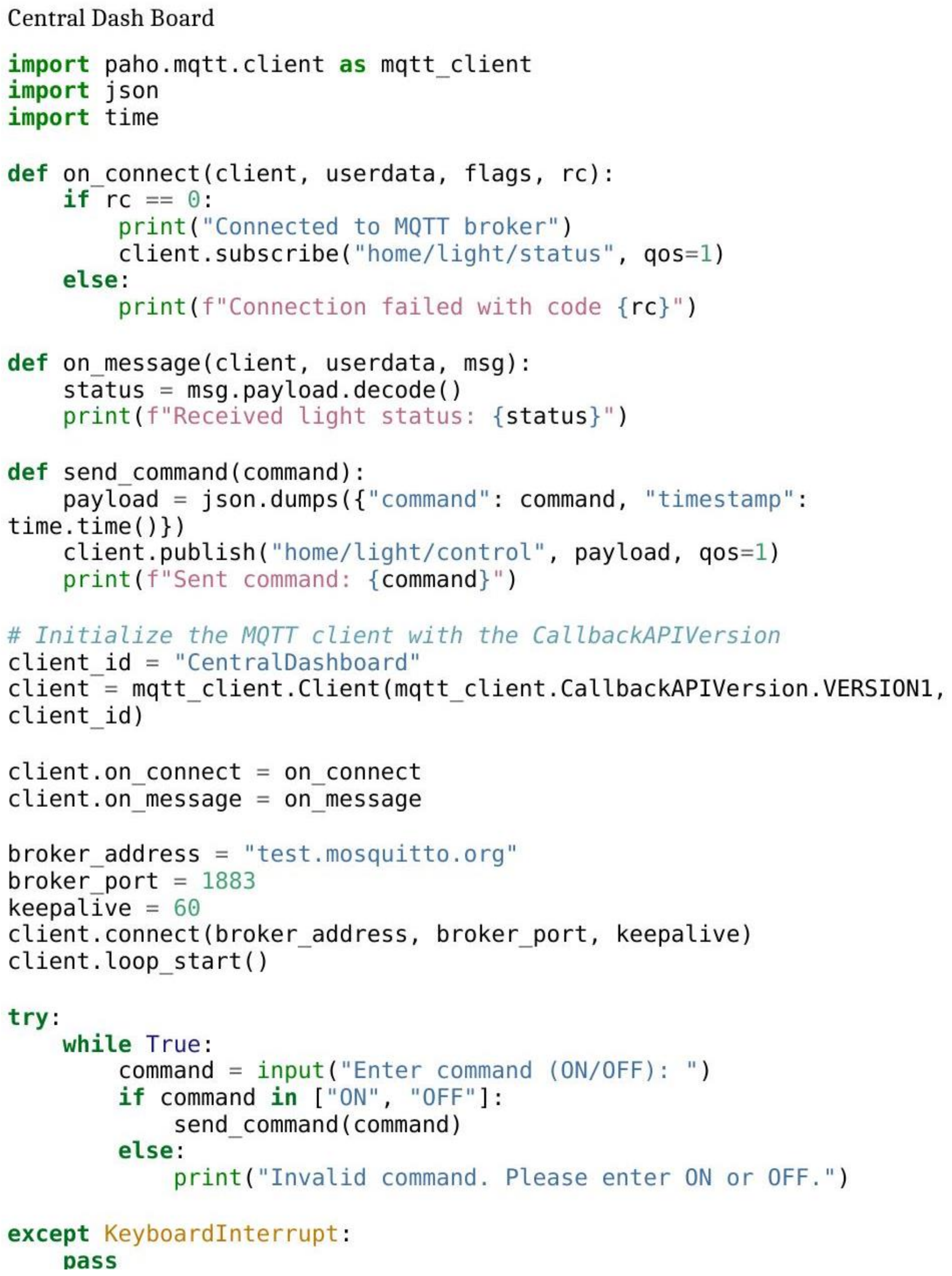
Turning Off the Light:

The Central Hub sends a "TURN OFF" command to the Light Controller through the MQTT broker.

The Light Controller processes the command, turns off the light bulb, and publishes a status update indicating that the light is now OFF.

The Central Hub receives this status update and shows the light's current state as OFF.

Code for Central Dashboard

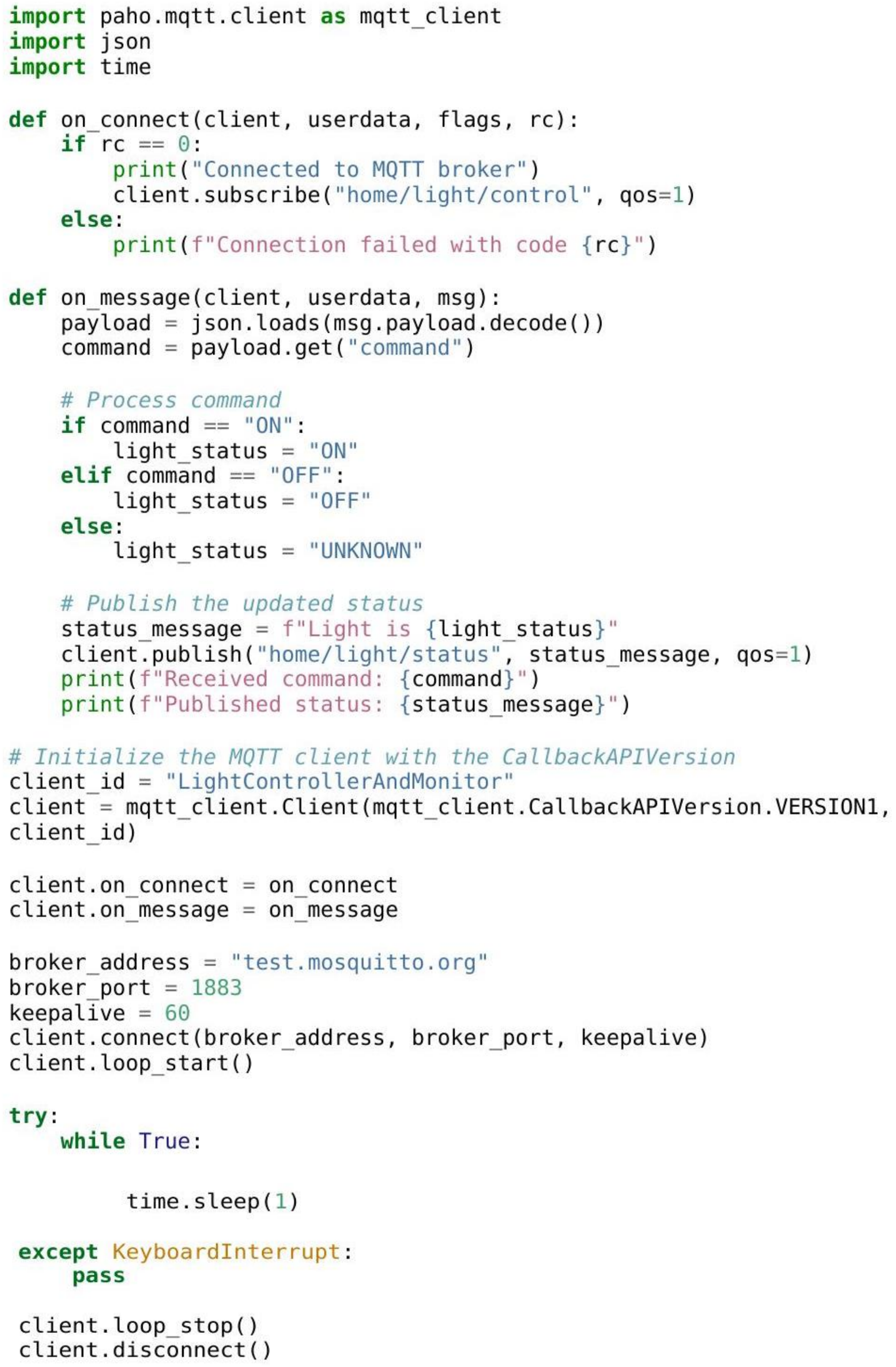


Central Dash Board

```
import paho.mqtt.client as mqtt_client
import json
import time

def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT broker")
        client.subscribe("home/light/status", qos=1)
    else:
        print(f"Connection failed with code {rc}")

def on_message(client, userdata, msg):
    status = msg.payload.decode()
    print(f"Received light status: {status}")

def send_command(command):
    payload = json.dumps({"command": command, "timestamp":
time.time()})
    client.publish("home/light/control", payload, qos=1)
    print(f"Sent command: {command}")

# Initialize the MQTT client with the CallbackAPIVersion
client_id = "CentralDashboard"
client = mqtt_client.Client(mqtt_client.CallbackAPIVersion.VERSION1,
client_id)

client.on_connect = on_connect
client.on_message = on_message

broker_address = "test.mosquitto.org"
broker_port = 1883
keepalive = 60
client.connect(broker_address, broker_port, keepalive)
client.loop_start()

try:
    while True:
        command = input("Enter command (ON/OFF): ")
        if command in ["ON", "OFF"]:
            send_command(command)
        else:
            print("Invalid command. Please enter ON or OFF.")

except KeyboardInterrupt:
    pass
```

```
client.loop_stop()
client.disconnect()
```

Code for Light Controller and Status Monitor



```
import paho.mqtt.client as mqtt_client
import json
import time

def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to MQTT broker")
        client.subscribe("home/light/control", qos=1)
    else:
        print(f"Connection failed with code {rc}")

def on_message(client, userdata, msg):
    payload = json.loads(msg.payload.decode())
    command = payload.get("command")

    # Process command
    if command == "ON":
        light_status = "ON"
    elif command == "OFF":
        light_status = "OFF"
    else:
        light_status = "UNKNOWN"

    # Publish the updated status
    status_message = f"Light is {light_status}"
    client.publish("home/light/status", status_message, qos=1)
    print(f"Received command: {command}")
    print(f"Published status: {status_message}")

# Initialize the MQTT client with the CallbackAPIVersion
client_id = "LightControllerAndMonitor"
client = mqtt_client.Client(mqtt_client.CallbackAPIVersion.VERSION1,
client_id)

client.on_connect = on_connect
client.on_message = on_message

broker_address = "test.mosquitto.org"
broker_port = 1883
keepalive = 60
client.connect(broker_address, broker_port, keepalive)
client.loop_start()

try:
    while True:

         time.sleep(1)

 except KeyboardInterrupt:
     pass

 client.loop_stop()
 client.disconnect()
```