



Universidad Rey Juan Carlos

E.T.S. INGENIERÍA DE TELECOMUNICACIÓN

# REDES DE ORDENADORES PARA ROBOTS Y MÁQUINAS INTELIGENTES

*Práctica 7*

Autor:  
Javier Izquierdo Hernández

April 26, 2024

# Contenidos

|           |  |           |
|-----------|--|-----------|
| <b>1</b>  | <b>Consideraciones iniciales</b>                           | <b>2</b>  |
| <b>2</b>  | <b>Usuarios con linux en su PC o utilizando una imagen</b> | <b>3</b>  |
| 2.1       | Instalación de mqtttx . . . . .                            | 3         |
| <b>3</b>  | <b>Usuarios con vncweb o usuarios en los laboratorios</b>  | <b>6</b>  |
| 3.1       | Uso de mqtttx . . . . .                                    | 6         |
| <b>4</b>  | <b>Connect</b>   | <b>8</b>  |
| <b>5</b>  | <b>Ping</b>  | <b>13</b> |
| <b>6</b>  | <b>Subscribe</b>   | <b>14</b> |
| <b>7</b>  | <b>Publish</b>   | <b>16</b> |
| <b>8</b>  | <b>Subscribe-Publish-Qos</b>                               | <b>19</b> |
| 8.1       | Subscripción con Qos=0 . . . . .                           | 19        |
| 8.2       | Subscripción con Qos=2 . . . . .                           | 21        |
| <b>9</b>  | <b>Retain</b>  | <b>24</b> |
| <b>10</b> | <b>Unsubscribe</b>   | <b>27</b> |
| <b>11</b> | <b>Last Will y Filtros de topic</b>                        | <b>28</b> |
| <b>12</b> | <b>Persistencia sesión</b>                                 | <b>30</b> |

# 1 Consideraciones iniciales

Para realizar esta práctica se necesita utilizar software mqtttx que actúe como servidor mqtttx. Tienes 2 opciones: 1. Puedes instalar el software mqtttx en una máquina virtual con Ubuntu. Continúa en el paso 2.

2. Puedes utilizar el software mqtttx instalado en los laboratorios. Continúa en el paso 3.

## 2 Usuarios con linux en su PC o utilizando una imagen

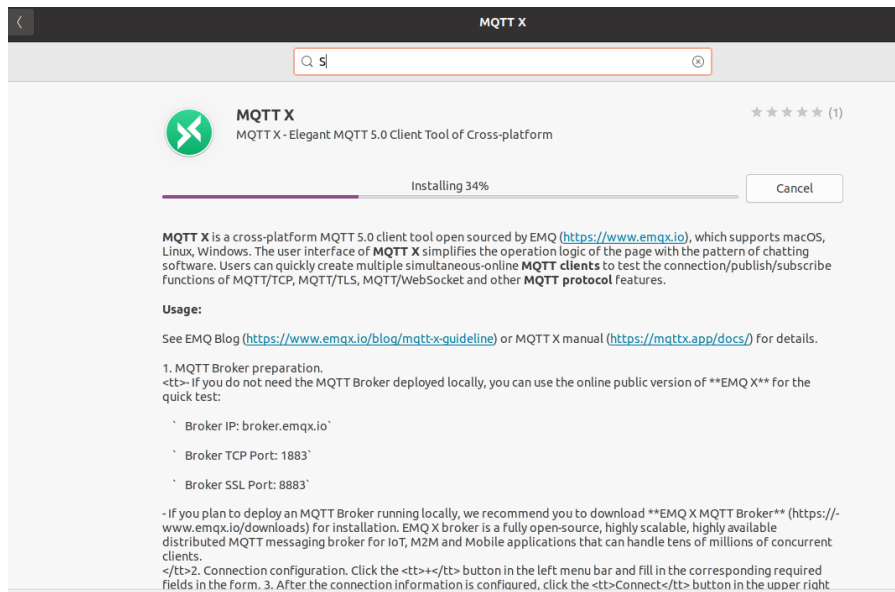
Importante: usuarios que utilicen su propio PC con Linux o tengan una imagen como la proporcionada aquí <https://labs.etsit.urjc.es/index.php/tutoriales/imagenes-ubuntu-virtualbox/>

### 2.1 Instalación de mqtttx

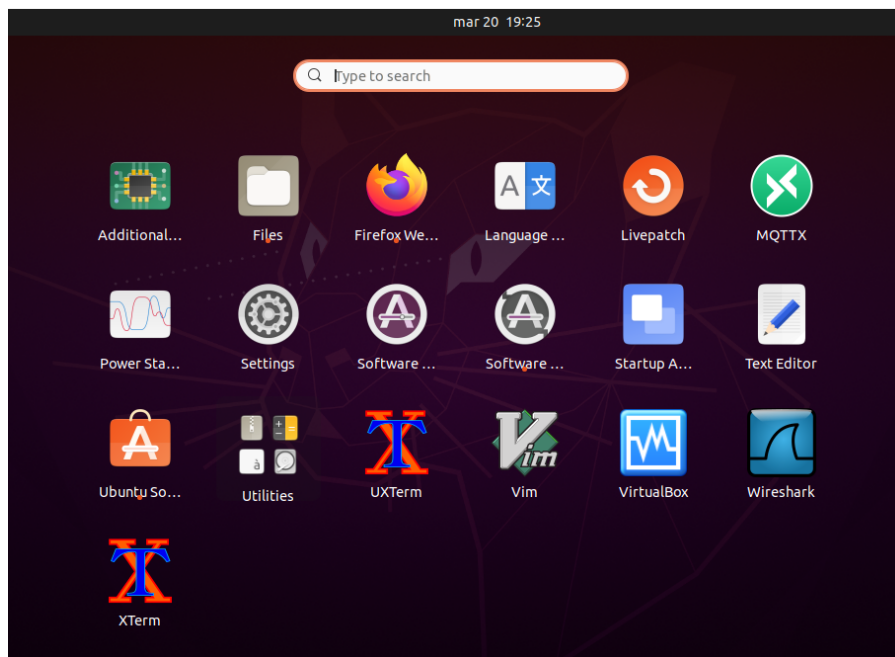
Instala la siguiente aplicación que será un cliente mqtt. Se utilizará la aplicación mqtttx, <https://mqtttx.app/>. La instalación en ubuntu se realiza ejecutando los siguientes comandos:

```
sudo apt update
sudo apt install snapd
sudo snap install mqtttx
```

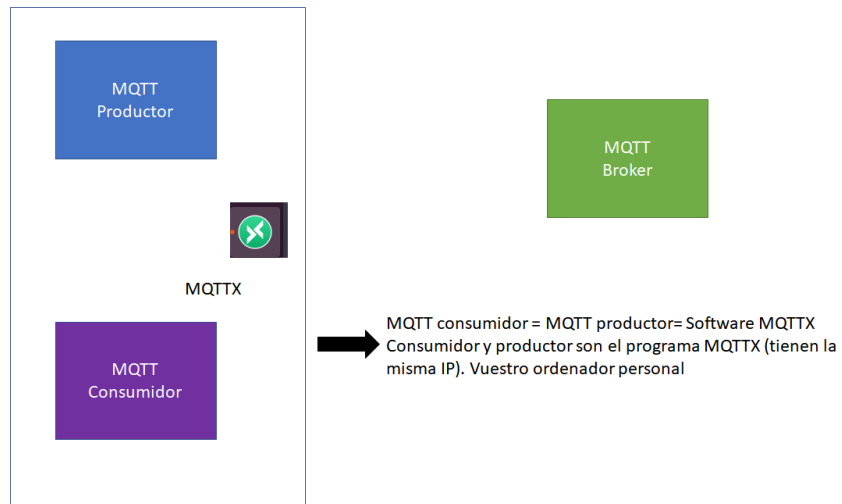
Si lo instaláis usando la interfaz gráfica se mostrará una pantalla como esta



Una vez instalada, se puede ejecutar de la siguiente forma:



**Importante:** En esta práctica, el consumidor y el productor están en el mismo host (vuestro ordenador) que ejecuta el cliente mqtt con el software (mqtttx)



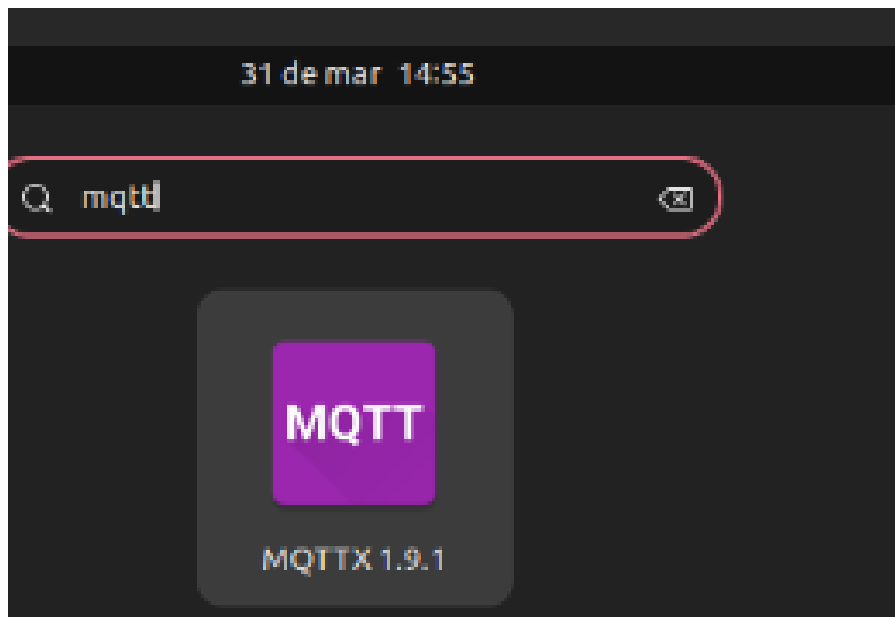
Continúa en el paso 4

## 3 Usuarios con vncweb o usuarios en los laboratorios

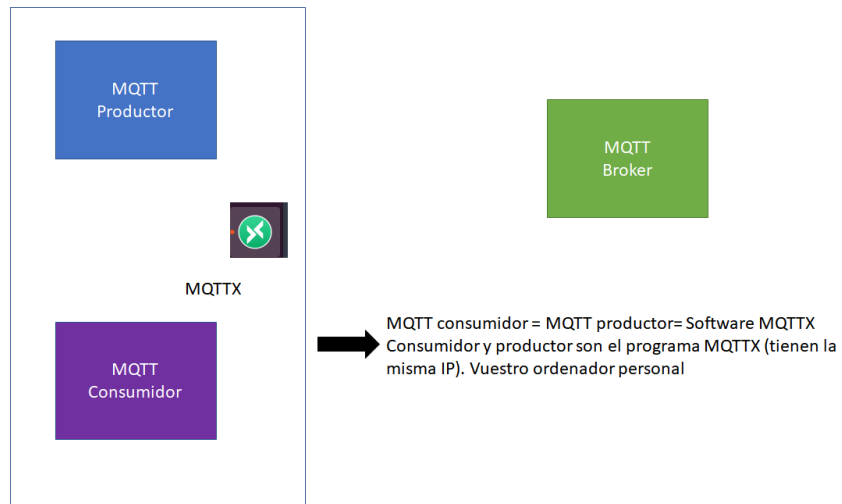
Conectate con tu usuario al laboratorio <https://labs.eif.urjc.es/vnc/> o en el pc de los laboratorios

### 3.1 Uso de mqtttx

Usa la aplicación instalada



**Importante:** En esta práctica, el consumidor y el productor están en el mismo host (vuestro ordenador) que ejecuta el cliente mqtt con el software (mqtttx)





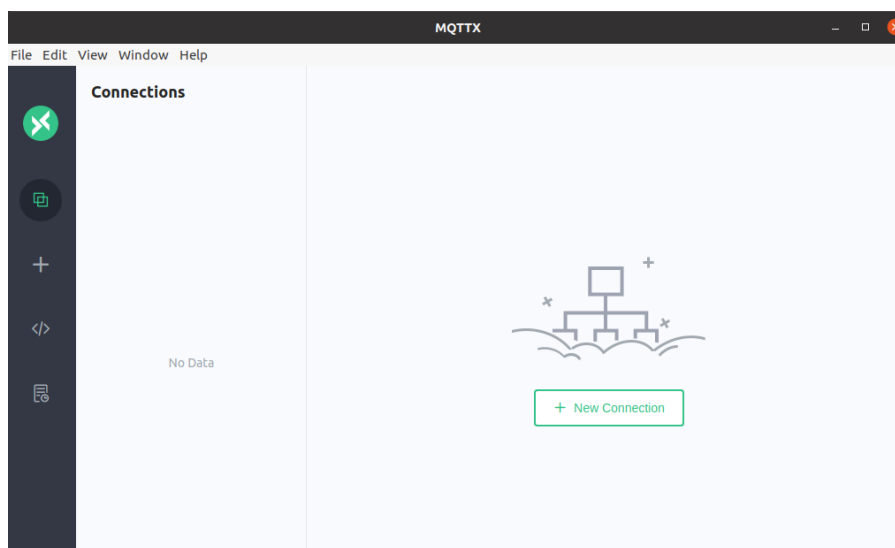
## 4 Connect

1. En un terminal ejecuta el siguiente comando para lanzar una captura de tráfico.

```
sudo tcpdump -i any -s 0 -w mqtt-01.cap
```

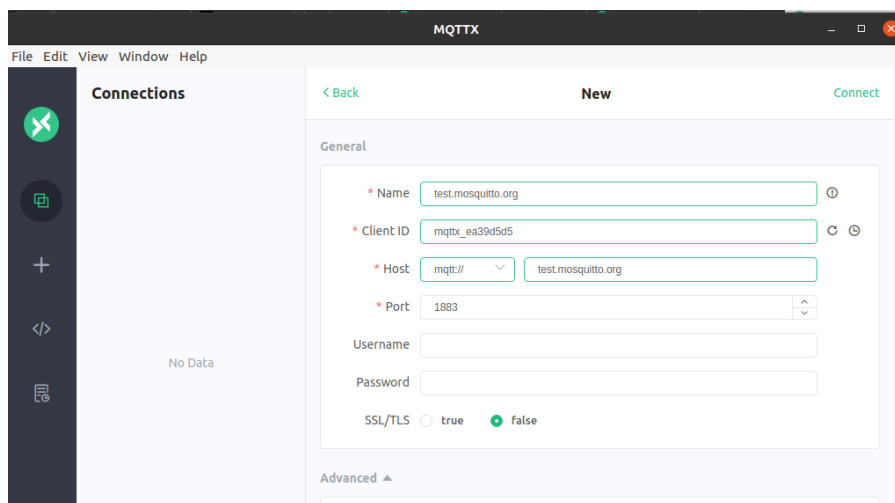
Nota: Puedes capturar paquetes también con wireshark seleccionando any como la interfaz que quieres capturar. Para ello tienes que haber instalado wireshark con permisos para que cualquier usuario pueda capturar tráfico, o deberás lanzar wireshark con sudo.

2. Crea una nueva conexión

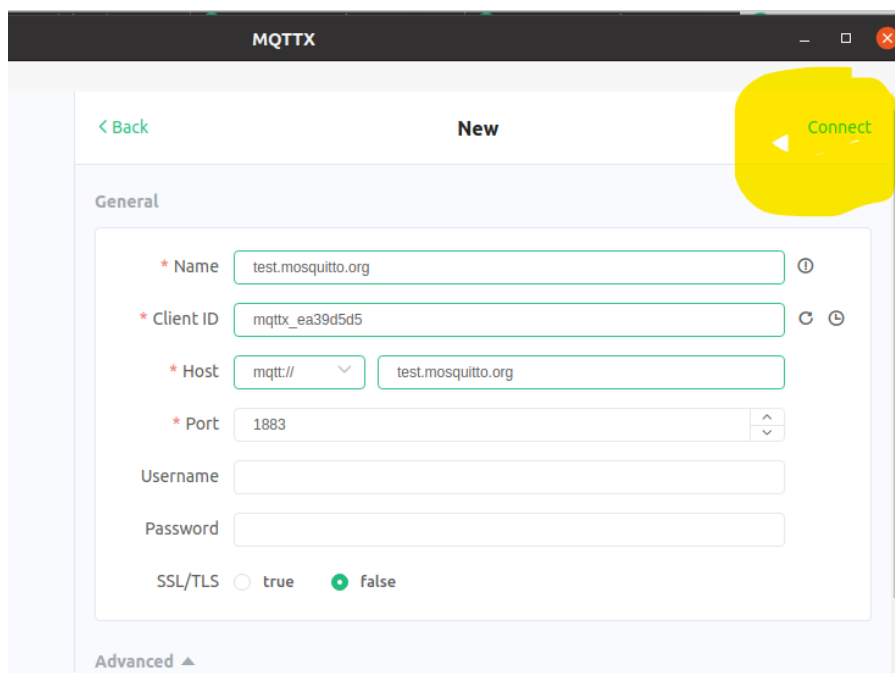


Con las siguientes propiedades

Name: test.mosquitto.org Host: test.mosquitto.org Port:1883



3. Conectate usando el botón del navegador Connect (arriba a la derecha)



Verifica que la versión es 3.1 y que el auto reconnect está deshabilitado. Este menú esta disponible en la parte Advanced de la configuración del servidor

Advanced ▲

MQTT Version

Connect Timeout

Keep Alive

Auto Reconnect ☐

Clean Session ☐

La conexión se pondrá en verde:



4. Para la captura con Control+C.
5. Abre la captura en wireshark. Si no tienes permisos para abrir la captura, cámbiaselos con `sudo chmod`.  
La conexión de este MQTT cliente se realiza utilizando el puerto 1883. En algunas versiones de wirehark no se identifica automáticamente el protocolo MQTT. Si te ocurre así, localiza en la captura los paquetes que tienen el puerto 1883, por ejemplo con el filtro `tcp.port==1883`

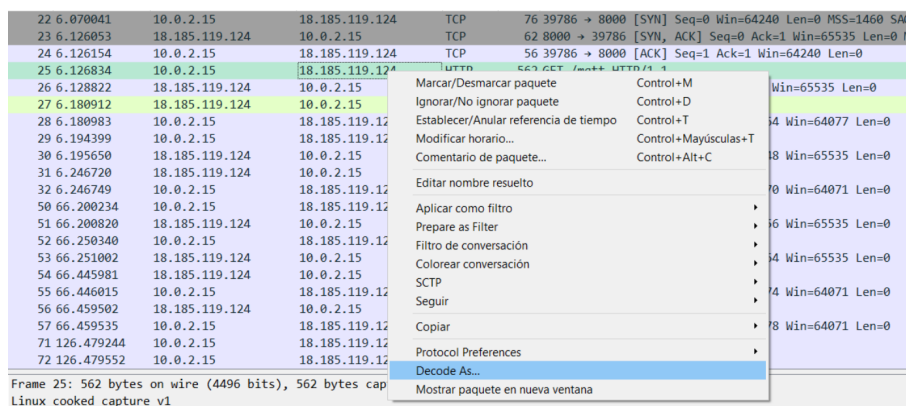
mqttx.cap

Archivo Edición Visualización Ir Captura Analizar Estadísticas Telefonía Wireless Herramientas Ayuda

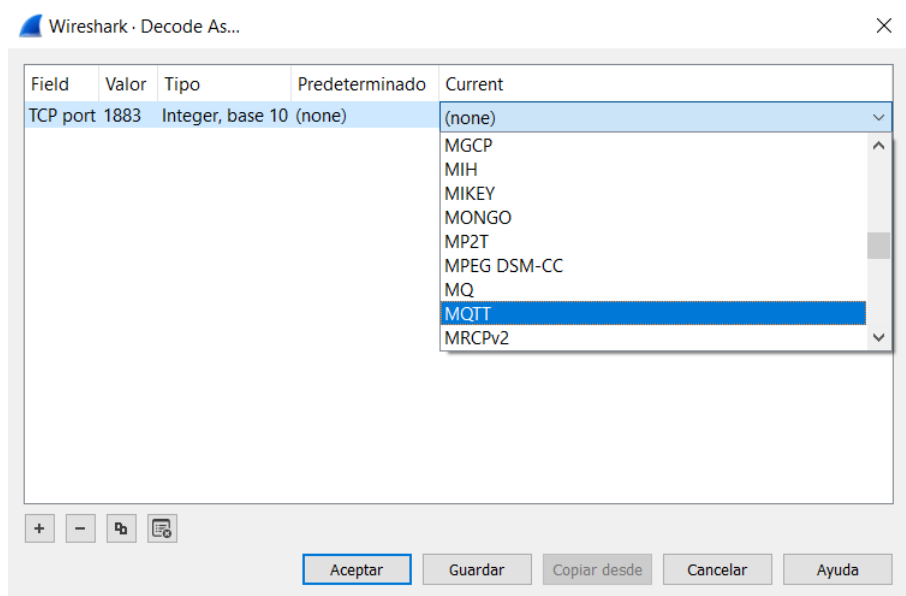
tcp.port == 1883

| No. | Time       | Source       | Destination  | Protocol | Length | Info  |
|-----|------------|--------------|--------------|----------|--------|---|
| 25  | 45.841278  | 10.0.2.15    | 52.36.58.151 | TCP      | 76     | 41818 → 1883 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 |
| 26  | 46.119551  | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460  |
| 27  | 46.119622  | 10.0.2.15    | 52.36.58.151 | TCP      | 56     | 41818 → 1883 [ACK] Seq=1 Ack=1 Win=64240 Len=0                |
| 28  | 46.121130  | 10.0.2.15    | 52.36.58.151 | MQTT     | 84     | Connect Command   |
| 29  | 46.121807  | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [ACK] Seq=1 Ack=29 Win=65535 Len=0               |
| 30  | 46.293457  | 52.36.58.151 | 10.0.2.15    | MQTT     | 62     | Connect Ack   |
| 31  | 46.293510  | 10.0.2.15    | 52.36.58.151 | TCP      | 56     | 41818 → 1883 [ACK] Seq=29 Ack=5 Win=64236 Len=0               |
| 75  | 106.300140 | 10.0.2.15    | 52.36.58.151 | MQTT     | 58     | Ping Request  |
| 76  | 106.300770 | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [ACK] Seq=5 Ack=31 Win=65535 Len=0               |
| 77  | 106.537595 | 52.36.58.151 | 10.0.2.15    | MQTT     | 62     | Ping Response   |
| 78  | 106.537645 | 10.0.2.15    | 52.36.58.151 | TCP      | 56     | 41818 → 1883 [ACK] Seq=31 Ack=7 Win=64234 Len=0               |
| 122 | 155.508425 | 10.0.2.15    | 52.36.58.151 | MQTT     | 109    | Publish Message [testtopic/#]                                 |
| 123 | 155.508928 | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [ACK] Seq=7 Ack=84 Win=65535 Len=0               |
| 124 | 155.690647 | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [FIN, ACK] Seq=7 Ack=84 Win=65535 Len=0          |
| 125 | 155.694072 | 10.0.2.15    | 52.36.58.151 | TCP      | 56     | 41818 → 1883 [FIN, ACK] Seq=84 Ack=8 Win=64233 Len=0          |
| 126 | 155.694627 | 52.36.58.151 | 10.0.2.15    | TCP      | 62     | 1883 → 41818 [ACK] Seq=8 Ack=85 Win=65535 Len=0               |

Selecciona uno de los mensajes y haciendo click con el botón derecho, Selecciona Decode as...:



Posteriormente indica que el protocolo será MQTT:



Una vez que wireshark identifica correctamente los paquetes del protocolo MQTT, para seleccionar los paquetes en la captura puedes usar simplemente un filtro con el nombre del protocolo: `mqtt`.  
Identifica el mensaje **Connect**.

|              |              |      |                     |
|--------------|--------------|------|---------------------|
| 10.1.156.160 | 91.121.93.94 | MQTT | 102 Connect Command |
| 91.121.93.94 | 10.1.156.160 | MQTT | 76 Connect Ack      |

6. ¿Qué dirección IP utiliza el cliente MQTT?

10.1.156.160

7. ¿Qué dirección IP utiliza el broker?

91.121.93.94

8. ¿Qué versión se utiliza del protocolo MQTT?

v3.1

9. ¿Qué client id utiliza?

mqtttx\_4964f260

10. En el mensaje de Connection Ack, ¿qué código devuelve? ¿qué significa ese código?

Devuelve el código 0 que significa conexión aceptada.

11. ¿Tiene el parámetro Clean Session activo?

No

## 5 Ping

1. Ponte a capturar tráfico en el fichero [mqtt-02.cap](#).
2. Verifica que la conexión está activa. Deja 3 minutos la captura.
3. Para la captura con Control+C.
4. Abre la captura en wireshark. Identifica los mensajes Ping
5. ¿Cada cuánto tiempo se producen los mensajes Ping? ¿Dónde has configurado ese valor? ¿Dónde lo puedes observar? Pista: Chequea los mensajes connect del apartado anterior

Se mandan cada 60 segundos o 1 minuto, y esto esta configurado en el campo Keep Alive que tiene un valor de 60, es decir que la conexión se restaura cada 60 segundos al mandar el ping para comprobar si todo sigue igual.

|              |              |      |                  |
|--------------|--------------|------|------------------|
| 10.1.156.160 | 91.121.93.94 | MQTT | 74 Ping Request  |
| 91.121.93.94 | 10.1.156.160 | MQTT | 74 Ping Response |
| 10.1.156.160 | 91.121.93.94 | MQTT | 74 Ping Request  |
| 91.121.93.94 | 10.1.156.160 | MQTT | 74 Ping Response |
| 10.1.156.160 | 91.121.93.94 | MQTT | 74 Ping Request  |
| 91.121.93.94 | 10.1.156.160 | MQTT | 74 Ping Response |
| 10.1.156.160 | 91.121.93.94 | MQTT | 74 Ping Request  |
| 91.121.93.94 | 10.1.156.160 | MQTT | 74 Ping Response |

6. ¿Qué longitud de mensaje tiene el ping request?

0 bytes.

7. ¿Qué longitud de mensaje tiene el ping response?

0 bytes.

## 6 Subscribe

1. Ponte a capturar tráfico en el fichero [mqtt-03.cap](#).
2. Verifica que la conexión está activa.
3. Subscribe tu cliente (consumidor) al siguiente tema con Qos 0.  
**IMPORTANTE: Utiliza como X el valor que tienen tus direcciones IP en los escenarios de red las prácticas, como por ejemplo, el segundo byte de las direcciones IP en la práctica 4:**

testtopic/p7/X/Qos0

4. Subscribe tu cliente (consumidor) al siguiente tema con Qos 1. Asegúrate de ajustar el campo QoS de la subscripción.

testtopic/p7/X/Qos1

5. Subscribe tu cliente (consumidor) al siguiente tema con Qos 2. Asegúrate de ajustar el campo QoS de la subscripción.

testtopic/p7/X/Qos2

6. Para la captura con Control+C.
7. Abre la captura en wireshark. Identifica los mensajes **Subscribe**
8. ¿A qué topics te has suscrito?

Me he suscrito a testtopic/p7/209/Qos0, testtopic/p7/209/Qos1, testtopic/p7/209/Qos2.

9. Identifica los diferentes mensaje de subscripción por tu cliente como Qos=0, Qos=1, Qos=2 ¿cómo los has reconocido?  
Chequéalo identificando el contenido del mensaje.

Por el campo Requested QoS

```

> Frame 618: 100 bytes on wire (800 bits), 100 bytes captured (800 bits)
> Linux cooked capture v2
> Internet Protocol Version 4, Src: 10.1.156.160, Dst: 91.121.93.94
> Transmission Control Protocol, Src Port: 50972, Dst Port: 1883, Seq: 3, Ack: 3, Len: 28
- MQ Telemetry Transport Protocol, Subscribe Request
  > [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
  > Header Flags: 0x82, Message Type: Subscribe Request
    1000 .... = Message Type: Subscribe Request (8)
    .... 0010 = Reserved: 2
  Msg Len: 26
  Message Identifier: 7675
  Topic Length: 21
  Topic: testtopic/p7/209/Qos0
  Requested QoS: At most once delivery (Fire and Forget) (0)
```

10. ¿Quién es el originante (productor del mensaje)? ¿Cliente o Broker?

El originante de los mensajes de subscripción es el cliente.

11. Identifica los mensajes Subscribe Ack. ¿Qué campo utiliza para identificar el topic del Subscribe Request en cada uno de los mensajes ¿Cuántos mensajes Subscribe tienes? ¿Cuántos mensajes Subscribe Ack?

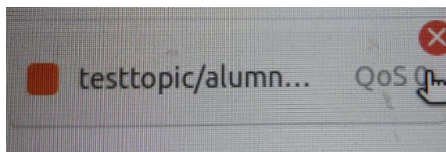
Los mensajes de Subscribe Ack usan el campo message identifier para saber que mensaje contiene el topic que aceptan.

Por último, como hay 3 Subscribe hay 3 Ack.

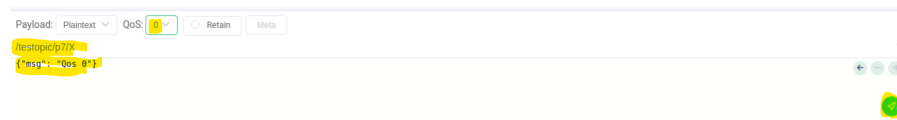


## 7 Publish

1. Elimina las subscripciones del apartado anterior:



2. Ponte a capturar tráfico en el fichero [mqtt-04.cap](#).
3. Verifica que la conexión está activa.
4. Publica un mensaje con Qos 0 con el topic testtopic/p7/X. Incluye en el texto del mensaje Qos 0



NOTA: Verás aparecer los mensajes que publicas justo encima de donde los escribes. También puedes comprobar el comportamiento del cliente en el panel de log:

```

Log
41 [2021-04-26 23:34:22] [INFO] MQTT client disconnect, client ID : mqttx_1/7991c
42 [2021-04-26 11:58:52] [INFO] Connect client, MQTT/TCP connection: mqtt://test.mosquitto.org:1883
43 [2021-04-26 11:58:52] [INFO] MQTT client with ID 6c22601c-6205-4987-8083-bec3f801cb4 assigned
44 [2021-04-26 11:58:52] [INFO] Connect success, MQTT.js onConnect trigger
45 [2021-04-26 12:12:49] [INFO] Save subscription topic:testtopic/alumno/23/Qos0 succeeded
46 [2021-04-26 12:13:15] [INFO] Save subscription topic:testtopic/alumno/X/Qos1 succeeded
47 [2021-04-26 12:13:42] [INFO] Save subscription topic:testtopic/alumno/X/Qos2 succeeded
48 [2021-04-26 12:13:49] [INFO] Unsubscribe topic: testtopic/alumno/X/Qos1
49 [2021-04-26 12:14:13] [INFO] Save subscription topic:testtopic/alumno/23/Qos1 succeeded
50 [2021-04-26 12:14:22] [INFO] Unsubscribe topic: testtopic/alumno/X/Qos2
51 [2021-04-26 12:14:54] [INFO] Save subscription topic:testtopic/alumno/23/Qos2 succeeded
52 [2021-04-26 12:20:57] [INFO] Unsubscribe topic: testtopic/alumno/23/Qos0
53 [2021-04-26 12:20:58] [INFO] Unsubscribe topic: testtopic/alumno/23/Qos1
54 [2021-04-26 12:20:59] [INFO] Unsubscribe topic: testtopic/alumno/23/Qos2
55 [2021-04-26 12:24:26] [INFO] Successfully published message "{\n  \"msg\": \"Qos 0\"\n}" to topic "/testtopic/alumno/X"
56 [2021-04-26 12:24:29] [INFO] Successfully published message "{\n  \"msg\": \"Qos 0\"\n}" to topic "/testtopic/alumno/X"
57 [2021-04-26 12:24:52] [INFO] Successfully published message "{\n  \"msg\": \"Qos 1\"\n}" to topic "/testtopic/alumno/X"
58 [2021-04-26 12:24:53] [INFO] Successfully published message "{\n  \"msg\": \"Qos 1\"\n}" to topic "/testtopic/alumno/X"
59 [2021-04-26 12:24:59] [INFO] Successfully published message "{\n  \"msg\": \"Qos 1\"\n}" to topic "/testtopic/alumno/X"
60 [2021-04-26 12:25:33] [INFO] Successfully published message "{\n  \"msg\": \"Qos 2\"\n}" to topic "/testtopic/alumno/X"
61 [2021-04-26 12:25:41] [INFO] Successfully published message "{\n  \"msg\": \"Qos 1\"\n}" to topic "/testtopic/alumno/X"
62 [2021-04-26 12:25:58] [INFO] Successfully published message "{\n  \"msg\": \"Qos 0\"\n}" to topic "/testtopic/alumno/X"
63 [2021-04-26 12:30:23] [INFO] Successfully published message "{\n  \"msg\": \"prueba\"\n}" to topic "testtopic/alumno/prueba"
64 [2021-04-26 12:30:35] [INFO] Successfully published message "{\n  \"msg\": \"prueba\"\n}" to topic "testtopic/alumno/prueba"
65

```

- Publica un mensaje con Qos 1 con el topic testtopic/p7/X. Incluye en el texto del mensaje Qos 1.
- Publica un mensaje con Qos 2 con el topic testtopic/p7/X. Incluye en el texto del mensaje Qos 2.
- Para la captura con Control+C.
- Abre la captura en wireshark Identifica los mensajes **Publish**

|              |              |      |  |
|--------------|--------------|------|--|
| 10.1.156.160 | 91.121.93.94 | MQTT | 107 Publish Message [testtopic/p7/209]           |
| 10.1.156.160 | 91.121.93.94 | MQTT | 109 Publish Message (id=7681) [testtopic/p7/209] |
| 91.121.93.94 | 10.1.156.160 | MQTT | 76 Publish Ack (id=7681)                         |
| 10.1.156.160 | 91.121.93.94 | MQTT | 109 Publish Message (id=7682) [testtopic/p7/209] |
| 91.121.93.94 | 10.1.156.160 | MQTT | 78 Publish Received (id=7682)                    |
| 10.1.156.160 | 91.121.93.94 | MQTT | 76 Publish Release (id=7682)                     |
| 91.121.93.94 | 10.1.156.160 | MQTT | 76 Publish Complete (id=7682)                    |

- Identifica el mensaje publicado por tu cliente como Qos = 0 ¿Cómo lo has reconocido? ¿Hay algún mensaje Publish Ack?  
Verifica tu suposición mirando también el contenido del mensaje.

Lo he reconocido por el campo QoS level en la cabecera de MQTT.  
No hay ningún mensaje de Ack.

```

Frame 525: 107 bytes on wire (856 bits), 107 bytes captured (856 bits)
Linux cooked capture v2
Internet Protocol Version 4, Src: 10.1.156.160, Dst: 91.121.93.94
Transmission Control Protocol, Src Port: 50972, Dst Port: 1883, Seq: 3, Ack: 3, Len: 35
MQ Telemetry Transport Protocol, Publish Message
  [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
  Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)
  Msg Len: 33
  Topic Length: 16
  Topic: testtopic/p7/209
  Message: 7b226d7367223a2022516f5330227d

```

- Identifica el mensaje publicado por tu cliente como Qos = 1 ¿Cómo lo has reconocido? ¿Hay algún mensaje Publish Ack para el correspondiente Publish

Request? ¿Cómo se hace la asociación entre el Publish Request y el Publish Ack?

Verifica tu suposición mirando también el contenido del mensaje.

Lo he reconocido por el campo QoS level en la cabecera de MQTT.

Si que existen un mensaje Publish Ack, cuya asociación con el request es simplemente el identificador de mensaje que acepta el Ack.

```

  Frame 627: 109 bytes on wire (872 bits), 109 bytes captured (872 bits)
  Linux cooked capture v2
  Internet Protocol Version 4, Src: 10.1.156.160, Dst: 91.121.93.94
  Transmission Control Protocol, Src Port: 50972, Dst Port: 1883, Seq: 38, Ack: 3, Len: 37
  MQTT Telemetry Transport Protocol, Publish Message
    [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
    Header Flags: 0x32, Message Type: Publish Message, QoS Level: At least once delivery (Acknowledged deliver)
    Msg Len: 35
    Topic Length: 16
    Topic: testtopic/p7/209
    Message Identifier: 7681
    Message: 7b226d7367223a2022516f5331227d

```

11. Identifica el mensaje publicado Qos = 2 ¿Cómo lo has reconocido? ¿Hay algún mensaje Publish Ack para el correspondiente Publish Request? ¿Qué otros mensajes hay? ¿Cómo se hace la asociación entre el Publish Request y el Publish Ack? ¿Y con el Publish Release y Complete?

Lo he reconocido por el campo QoS level en la cabecera de MQTT.

No existe ningún un mensaje Publish Ack, sin embargo hay otros 3 mensajes. Para hacer la asociación entre el Request y el "Ack" el broker envía un Publish Received en vez del Ack. Cuando el cliente recibe este mensaje lo confirma mandando de vuelta el Publish Release, y el broker vuelve a confirmar este mensaje con un Publish Complete.

## 8 Subscribe-Publish-Qos

### 8.1 Subscripción con Qos=0

1. Ponte a capturar tráfico en el fichero [mqtt-05.cap](#).
2. Verifica que la conexión está activa.
3. Subscribe tu cliente (consumidor) al siguiente tema con Qos=0, siendo X el número de tus direcciones IP:

`testtopic/p7/X/#`

4. Publica desde tu cliente (productor) un mensaje con Qos 0 con el topic `testtopic/p7/X`, siendo X tu el número de tus direcciones IP. Incluye en el texto del mensaje: Qos 0
5. Publica un mensaje con Qos 1 con el topic `testtopic/p7/X`. Incluye en el texto del mensaje: Qos 1
6. Publica un mensaje con Qos 2 con el topic `testtopic/p7/X`. Incluye en el texto del mensaje: Qos 2
7. Para la captura con Control+C.
8. Abre la captura en wireshark. **NOTA: Ten en cuenta que al transmitirse los mensajes de MQTT dentro de una conexión TCP puede haber varios mensajes MQTT dentro del mismo paquete capturado.**
9. Verifica en el correspondiente mensaje MQTT que la subscripción hecha tiene un Qos=0. ¿Dónde lo has identificado?

Lo he identificado en el campo Requested QoS.

```

Frame 179: 97 bytes on wire (776 bits), 97 bytes captured (776 bits)
Linux cooked capture v2
Internet Protocol Version 4, Src: 10.1.156.160, Dst: 91.121.93.94
Transmission Control Protocol, Src Port: 50972, Dst Port: 1883, Seq: 1, Ack: 1, Len: 25
MQ Telemetry Transport Protocol, Subscribe Request
  [Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]
  Header Flags: 0x82, Message Type: Subscribe Request
    Msg Len: 23
    Message Identifier: 7683
    Topic Length: 18
    Topic: testtopic/p7/209/#
    Requested QoS: At most once delivery (Fire and Forget) (0)

```

10. Identifica los mensajes Publish enviados por el cliente y los enviados por el broker. ¿Cómo los has reconocido?

Los mensajes que envía el broker serán los que compartan la Ip de origen con el que envía el Ack o con el que el puerto de origen sea 1883.

11. Para el caso específico del mensaje publicado por tu cliente (productor) con un Qos=0, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)?

Se envía un mensaje de tipo Message desde el cliente y otro desde el broker

12. Para el caso específico del mensaje publicado por tu cliente (productor) con un Qos=1, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)? ¿Hay algún mensaje Publish Ack (cuántos)? ¿En qué momento temporal se transmite el mensaje de Publish desde el broker al consumidor (antes o después del mensaje Publish Ack)?

Se envía un mensaje de tipo Message desde el cliente y el broker envía uno de tipo Message y otro de tipo Ack.

El Publish Message se transmite antes del Ack.

13. Para el caso específico del mensaje publicado por tu cliente (productor) con un Qos=2, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)? ¿Hay algún mensaje Publish Ack (cuántos)? ¿Después de qué mensaje se transmite el mensaje de Publish desde el broker al consumidor?

El cliente envía un mensaje de tipo Message y otro de tipo Release, y el broker transmite un mensaje de tipo Received, otro de Message y otro de Complete. No hay ningún mensaje de Ack, sin embargo hay un Received y un Complete.

El mensaje de Publish Message lo envía el broker al recibir el mensaje de Release.

## 8.2 Subscripción con Qos=2

1. Ponte a capturar tráfico en el fichero [mqtt-06.cap](#).
2. Verifica que la conexión está activa.
3. **Elimina tu subscripción del cliente anterior**
4. Subscribe tu cliente (consumidor) al siguiente tema con Qos=2

testtopic/p7/X/#

5. Publica desde tu cliente (productor) un mensaje con Qos 0 con el topic testtopic/p7/X. Incluye en el texto del mensaje: Qos 0
6. Publica un mensaje con Qos 1 con el topic testtopic/p7/X. Incluye en el texto del mensaje: Qos 1
7. Publica un mensaje con Qos 2 con el topic testtopic/p7/X. Incluye en el texto del mensaje: Qos 2
8. Para la captura con Control+C.
9. Abre la captura en wireshark.
10. Verifica en el correspondiente mensaje MQTT que la subscripción hecha tiene un Qos=2. ¿Cómo lo has identificado?

Lo he identificado en el campo Requested QoS.

11. Identifica los mensajes Publish enviados por el cliente y los enviados por el broker. ¿Cómo los has reconocido?

Los mensajes que envía el broker serán los que compartan la Ip de origen con el que envía el Ack o con el que el puerto de origen sea 1883.

12. Para el caso específico del mensaje publicado por tu cliente (productor) con un Qos=0, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde

el broker a tu cliente (consumidor)? ¿Observas algún mensaje Publish Complete? ¿Por qué?

Se envía un mensaje de tipo Message desde el cliente y otro desde el broker. No hay ningún mensaje de Publish Complete porque el cliente cuando lo publica tiene QoS 0, y cuando recibe el mensaje la QoS del mensaje recibido sigue siendo 0, por lo tanto no necesita hacer el Complete o el Ack.

13. Para el caso específico del mensaje publicado por tu cliente (productor) con un QoS=1, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)? ¿Hay algún mensaje Publish Ack (cuántos)? ¿En qué momento temporal se transmite el mensaje de Publish desde el broker al consumidor (antes o después del mensaje Publish Ack)? ¿Observas algún mensaje Publish Complete? ¿Por qué?

El cliente envía un mensaje de tipo Message y otro de tipo Ack al igual que el broker.

El mensaje es publicado por el broker antes del Ack.

No hay ningún mensaje de Publish Complete porque el cliente cuando lo publica tiene QoS 1, y cuando recibe el mensaje la QoS del mensaje recibido sigue siendo 1, por lo tanto se trata el paquete por ambas partes como si todo fuera con QoS 1.

14. Para el caso específico del mensaje publicado por tu cliente (productor) con un QoS=2, ¿qué mensajes de tipo Publish se envían del cliente (productor del mensaje) al broker? ¿Qué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)?

Ambos envían mensajes de tipo Message, Release, Complete y Received, ya que tienen que tratar el paquete por ambas partes como QoS 2

Compara el comportamiento de la subscripción con QoS=0 con el comportamiento de la subscripción con QoS=2:

15. Para la subscripción de tipo QoS=0, ¿cuál es la QoS desde el productor al broker para cada uno de los mensajes Publish generados (QoS (0,1,2))? ¿Cuál es la QoS desde el broker al consumidor para cada uno de los mensajes Publish generados (QoS (0,1,2))? ¿Cuál sería la QoS productor-consumidor en cada caso desde un punto de vista global de la comunicación para cada uno de los

mensajes Publish generados (Qos (0,1,2))?

Desde el productor al broker serían: 0, 1 y 2

Desde el broker al consumidor sería: 0, 0 y 0

Por lo tanto el QoS global sería 0, 0 y 0.

16. Para la subscripción de tipo Qos=2, ¿cual es la Qos desde el productor al broker para cada uno de los mensajes Publish generados (Qos (0,1,2))? ¿cuál es la Qos desde el broker al consumidor para cada uno de los mensajes Publish generados (Qos (0,1,2))? ¿Cuál sería la Qos productor-consumidor en cada caso desde un punto vista global de la comunicación para cada uno de los mensajes Publish generados (Qos (0,1,2))?

Desde el productor al broker serían: 0, 1 y 2

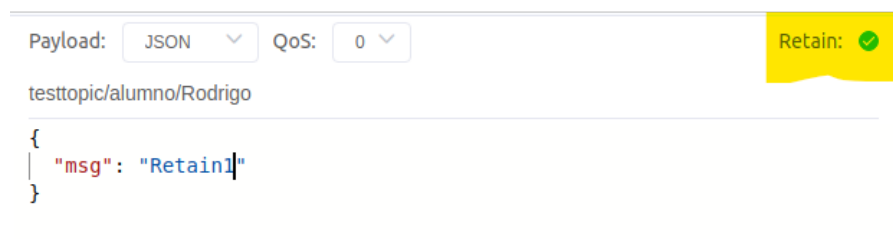
Desde el broker al consumidor sería: 2, 2 y 2

Por lo tanto el QoS global sería 0, 1 y 2.

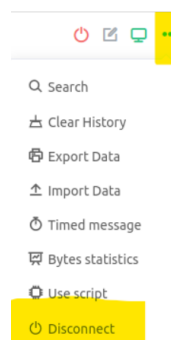


## 9 Retain

1. Ponte a capturar tráfico en el fichero [mqtt-07.cap](#).
2. Verifica que la conexión está activa.
3. Publica un mensaje con el topic `testtopic/p7/X` y Retain Activado con el Qos=0. Incluye en el texto del mensaje: `Retain1`



4. Publica un mensaje en el topic `testtopic/p7/X` y Retain Activado con el Qos=0. Incluye en el texto del mensaje: `Retain2`
5. Publica un mensaje en el topic `testtopic/p7/X` y Retain Desactivado con el Qos=0. Incluye en el texto del mensaje: `Retain3`
6. Desconecta la conexión.



7. Conéctate de nuevo
8. Suscríbete al mismo topic de antes: testtopic/p7/#
9. Para la captura con Control+C.
10. Abre la captura en wireshark. Identifica el mensaje Disconnect

|              |              |      |  |
|--------------|--------------|------|--|
| 10.1.156.160 | 91.121.93.94 | NQTT | 74 Ping Request                                      |
| 91.121.93.94 | 10.1.156.160 | NQTT | 74 Ping Response                                     |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 74 Disconnect Req                                    |
| 91.121.93.94 | 10.1.156.160 | NQTT | 102 Connect Command                                  |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Connect Ack                                       |
| 91.121.93.94 | 10.1.156.160 | NQTT | 97 Subscribe Request (id=45272) [testtopic/p7/209/#] |
| 10.1.156.160 | 91.121.93.94 | NQTT | 77 Subscribe Ack (id=45272)                          |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |

11. ¿Qué longitud tiene el mensaje?

Tiene longitud 0.

12. Identifica los mensajes que tienen el flag Retain en su cabecera Retain. ¿Qué mensajes son?

Son los siguientes mensajes

|              |              |      |  |
|--------------|--------------|------|--|
| 10.1.156.160 | 91.121.93.94 | NQTT | 74 Ping Request                                      |
| 91.121.93.94 | 10.1.156.160 | NQTT | 74 Ping Response                                     |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 91.121.93.94 | 10.1.156.160 | NQTT | 74 Disconnect Req                                    |
| 10.1.156.160 | 91.121.93.94 | NQTT | 102 Connect Command                                  |
| 91.121.93.94 | 10.1.156.160 | NQTT | 76 Connect Ack                                       |
| 10.1.156.160 | 91.121.93.94 | NQTT | 97 Subscribe Request (id=45272) [testtopic/p7/209/#] |
| 91.121.93.94 | 10.1.156.160 | NQTT | 77 Subscribe Ack (id=45272)                          |
| 10.1.156.160 | 91.121.93.94 | NQTT | 114 Publish Message [testtopic/p7/209]               |

13. ¿Qué texto de mensaje se recibe en el consumidor una vez que se vuelves a suscribir al topic testtopic/p7?

El mensaje de Retain2 porque es el último.

14. Ponte a capturar tráfico en el fichero mqtt-08.cap.
15. Verifica que la conexión está activa.
16. Publica un mensaje (como productor) con el topic testtopic/p7/X y Retain Activado y Qos=2. Incluye en el texto del mensaje Retain3 Qos2
17. Desconecta la conexión.

18. Conéctate de nuevo
19. Suscríbete al mismo topic de antes pero con Qos=2.
20. Para la captura con Control+C.
21. Abre la captura en wireshark.
22. ¿Cuántos mensajes Publish Message se envían con el flag retain activo? ¿Son los mensajes publicados por la misma IP? ¿Puedes identificar qué IP tiene el broker y cuál el cliente?

Se envían 3 mensajes con el flag activo, el primero proviene del cliente (cuando publicamos el mensaje) y el resto del broker.

|              |              |      |  |
|--------------|--------------|------|--|
| 10.1.156.160 | 91.121.93.94 | NQTT | 116 Publish Message (id=45275) [testtopic/p7/209]    |
| 91.121.93.94 | 10.1.156.160 | NQTT | 76 Publish Received (id=45275)                       |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Publish Release (id=45275)                        |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Publish Complete (id=45275)                       |
| 10.1.156.160 | 91.121.93.94 | NQTT | 74 Disconnect Req                                    |
| 91.121.93.94 | 10.1.156.160 | NQTT | 102 Connect Command                                  |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Connect Ack                                       |
| 91.121.93.94 | 10.1.156.160 | NQTT | 97 Subscribe Request (id=16405) [testtopic/p7/209/#] |
| 10.1.156.160 | 91.121.93.94 | NQTT | 77 Subscribe Ack (id=16405)                          |
| 91.121.93.94 | 10.1.156.160 | NQTT | 114 Publish Message [testtopic/p7/209]               |
| 10.1.156.160 | 91.121.93.94 | NQTT | 96 Unsubscribe Request (id=16406)                    |
| 91.121.93.94 | 10.1.156.160 | NQTT | 76 Unsubscribe Ack (id=16406)                        |
| 10.1.156.160 | 91.121.93.94 | NQTT | 97 Subscribe Request (id=16407) [testtopic/p7/209/#] |
| 91.121.93.94 | 10.1.156.160 | NQTT | 77 Subscribe Ack (id=16407)                          |
| 91.121.93.94 | 10.1.156.160 | NQTT | 116 Publish Message (id=3) [testtopic/p7/209]        |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Publish Received (id=3)                           |
| 91.121.93.94 | 10.1.156.160 | NQTT | 76 Publish Release (id=3)                            |
| 10.1.156.160 | 91.121.93.94 | NQTT | 76 Publish Complete (id=3)                           |

## 10 Unsubscribe

1. Ponte a capturar tráfico en el fichero [mqtt-09.cap](#).
2. Verifica que la conexión está activa.
3. Elimina la subscripción.
4. Para la captura con Control+C.
5. Abre la captura en wireshark. Identifica el mensaje Unsubscribe y Unsubscribe ACK

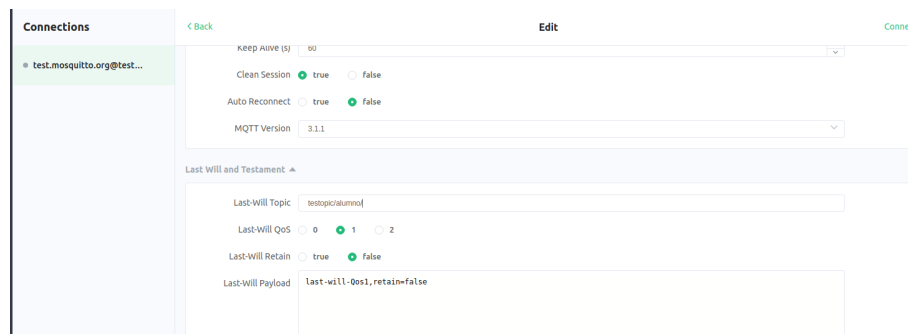
|               |               |      |                                   |
|---------------|---------------|------|-----------------------------------|
| 192.168.1.243 | 91.121.93.94  | MQTT | 96 Unsubscribe Request (id=46514) |
| 91.121.93.94  | 192.168.1.243 | MQTT | 76 Unsubscribe Ack (id=46514)     |

6. ¿De qué topic estás quitando la subscripción?

Del topic testtopic/p7/209/#.

# 11 Last Will y Filtros de topic

1. Ponte a capturar tráfico en el fichero [mqtt-10.cap](#).
2. Modifica la conexión para incluir los parámetro lastwill. Debes desconectar la conexión existente si esta activa.



3. Los parámetros son:

LastWillTopic=testtopic/p7/  
LastWill-Qos=1  
LastWill-Retain=False  
LastWill Message

Incluid el siguiente mensaje: LastWillQos1,RetainFalse

4. Verifica que la conexión está activa.
5. Elimina cualquier subscripción que tengas
6. Crea una subscripción testtopic/p7/X/+
7. Publica el siguiente topic: testtopic/p7/X/aa/bb
8. Publica el siguiente topic: testtopic/p7/X/aa

9. ¿Cuántos mensajes se reciben por parte del consumidor? ¿Por qué?

Solo 1 porque el comodín "+" hace que solo coincida un nivel de tema.

10. Crea una subscripción testtopic/p7/X/#

11. Publica el siguiente topic testtopic/p7/X/aa/bb

12. Publica el siguiente topic testtopic/p7/X/aa

13. ¿Cuántos mensajes se reciben por parte del consumidor? ¿Por qué?

Se reciben los 2 del nuevo subscriptor y 1 del anterior, porque el comodín del "#" coincide con cualquier número de niveles dentro de un tema.

14. Para la captura con Control+C.

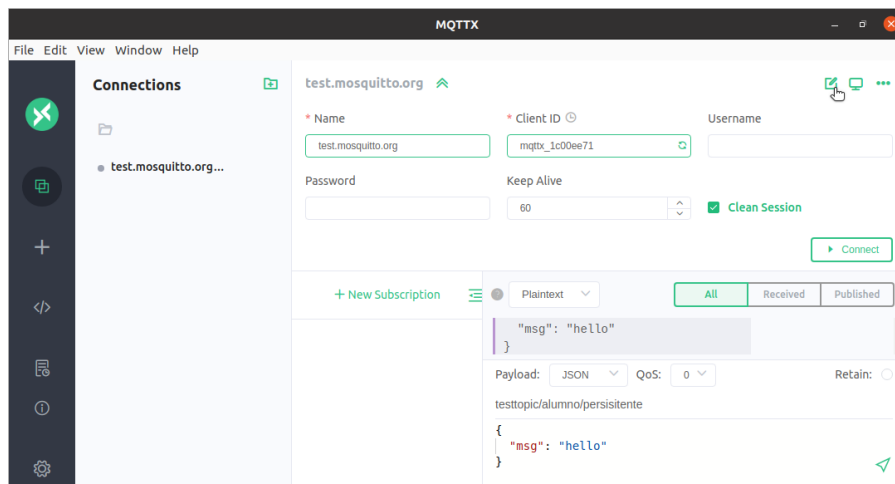
15. Abre la captura en wireshark

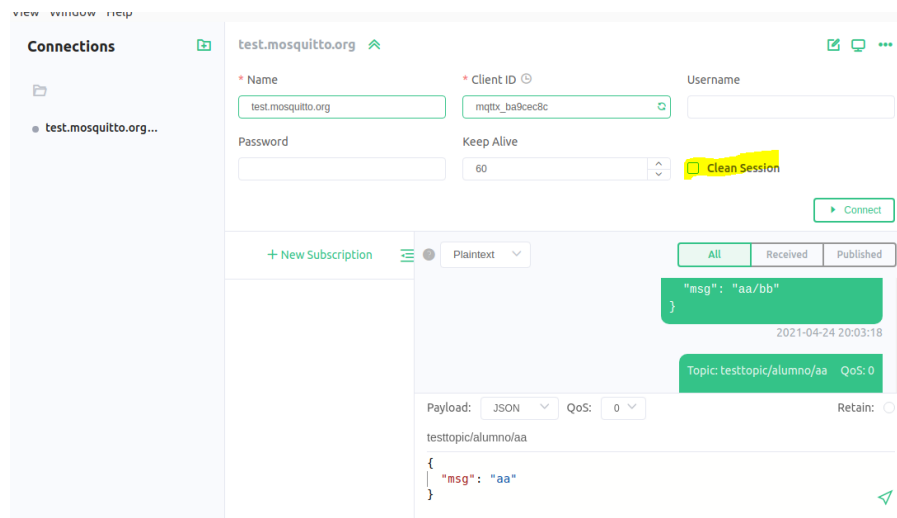
16. ¿En qué mensaje puedes ver el parámetro LastWill? ¿Qué lastWillQos tiene?

Solo se puede ver en el Connect y tiene un QoS de 1.

## 12 Persistencia sesión

1. Ponte a capturar tráfico creando el fichero [mqtt-11.cap](#).
2. Modifica la conexión para conectar usando el parámetro **CleanSession** activado (**true**)





3. Crea una subscripción testtopic/p7/X/#
4. Desconecta la sesión con el broker (servidor)
5. Conectate de nuevo con el parámetro **Clean Session Activado**
6. Publica el siguiente topic: testtopic/p7/X/ con Qos=2
7. Desconecta la conexión con el broker (servidor)
8. Modifica la conexión para conectar usando el parámetro **CleanFlag desactivado (false)**
9. Crea una subscripción testtopic/p7/X/#
10. Desconecta la sesión con el broker (servidor)
11. Conectate de nuevo con el parámetro **Clean Session desactivado**
12. Publica el siguiente topic: testtopic/p7/X/ con Qos=2
13. Desconecta la conexión
14. Para la captura.
15. Abre la captura en wireshark
16. ¿En el cliente consumidor has recibido algún mensaje cuando el Clean Session estaba habilitado? ¿y cuando estaba deshabilitado? ¿Por qué?



Cuando estaba el flag activado no he recibido ningún mensaje porque al cerrar la sesión el subscriber se ha desuscrito del topic.  
Mientras que cuando estaba deshabilitado he recibido los mensajes porque el subscriber sigue conectado a ese topic.