

# Redes de Ordenadores para Robots y Máquinas Inteligentes

## Práctica 7: IoT

GSyC

Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación  
URJC

Abril de 2024

### 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

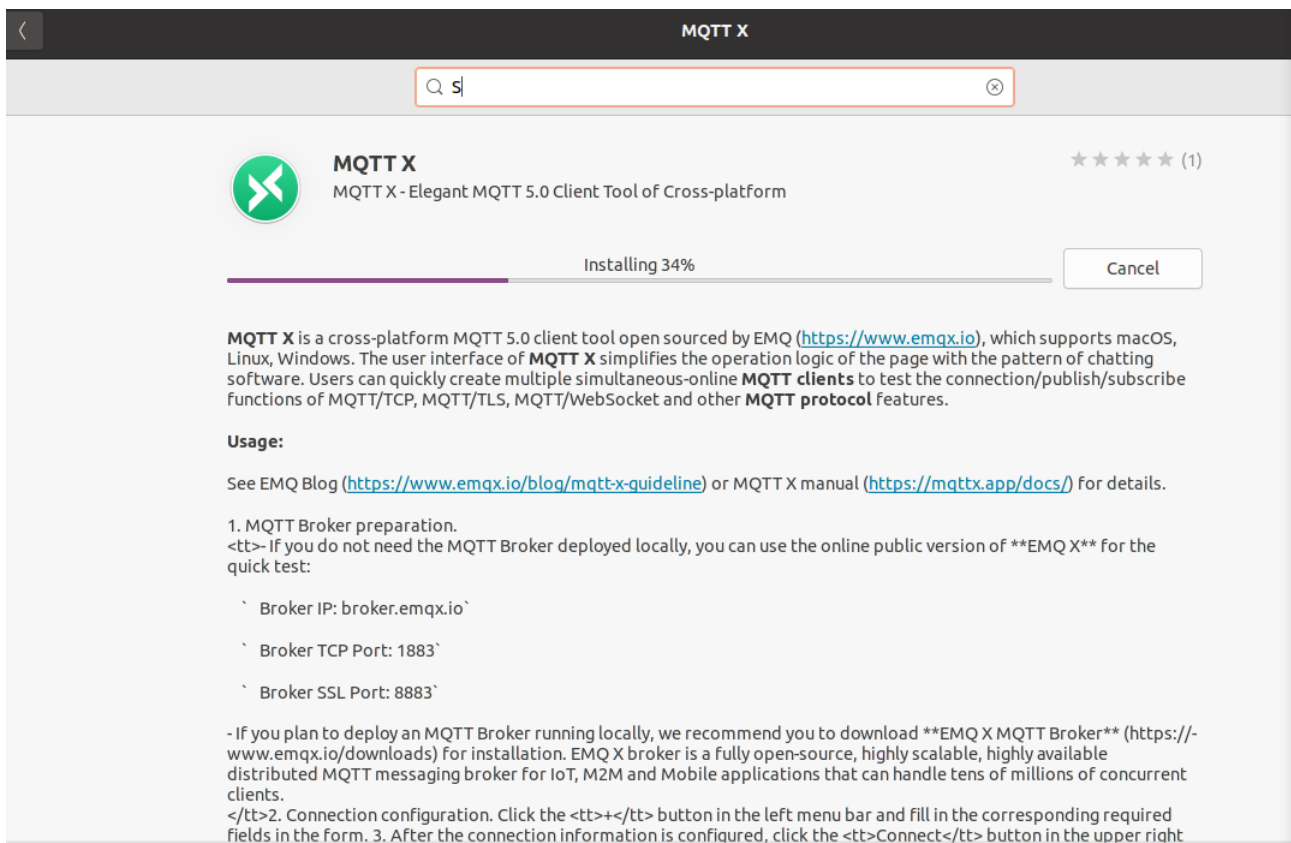
Importante: usuarios que utilicen su propio PC con Linux o tengan una imagen como la proporcionada aquí <https://labs.etsit.urjc.es/index.php/tutoriales/imagenes-ubuntu-virtualbox/>

#### 2.1. Instalación de mqtttx

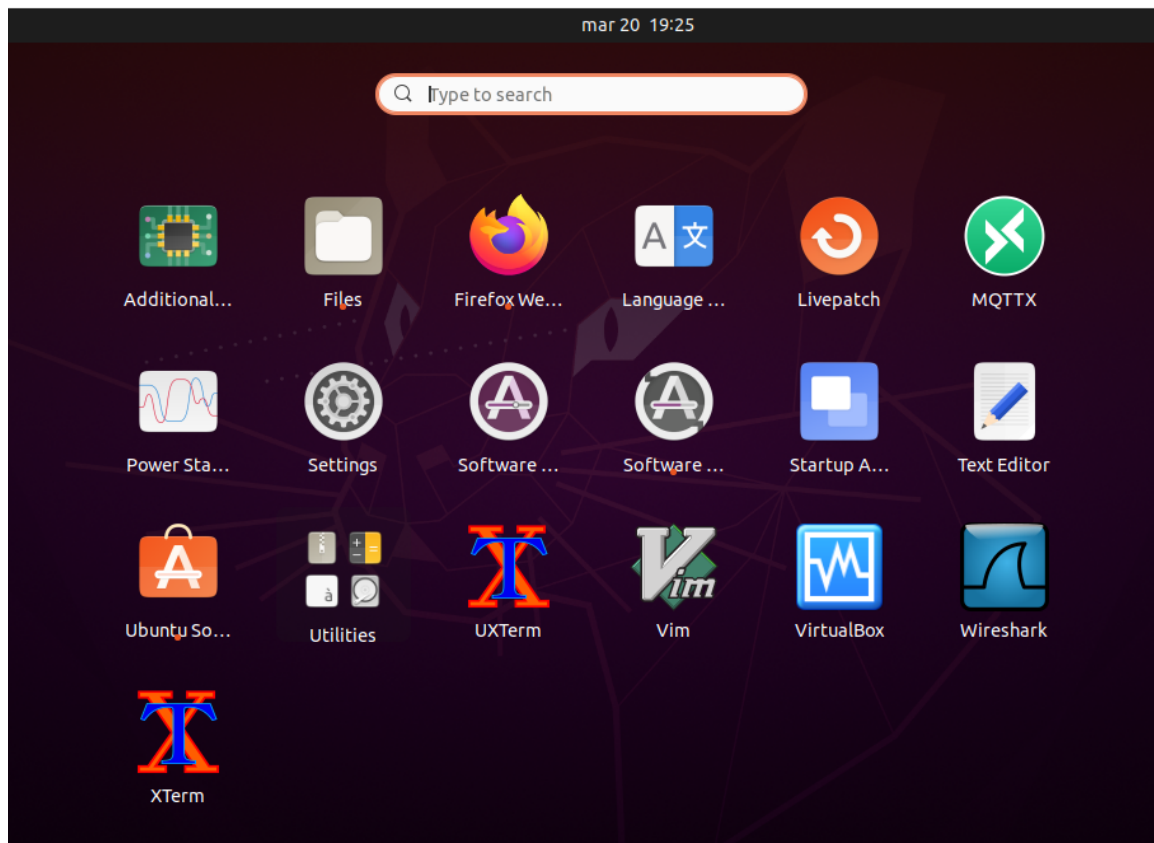
Instala la siguiente aplicación que será un cliente mqtt. Se utilizará la aplicación mqtttx, <https://mqtttx.app/>. La instalación en ubuntu se realiza ejecutando los siguientes comandos:

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

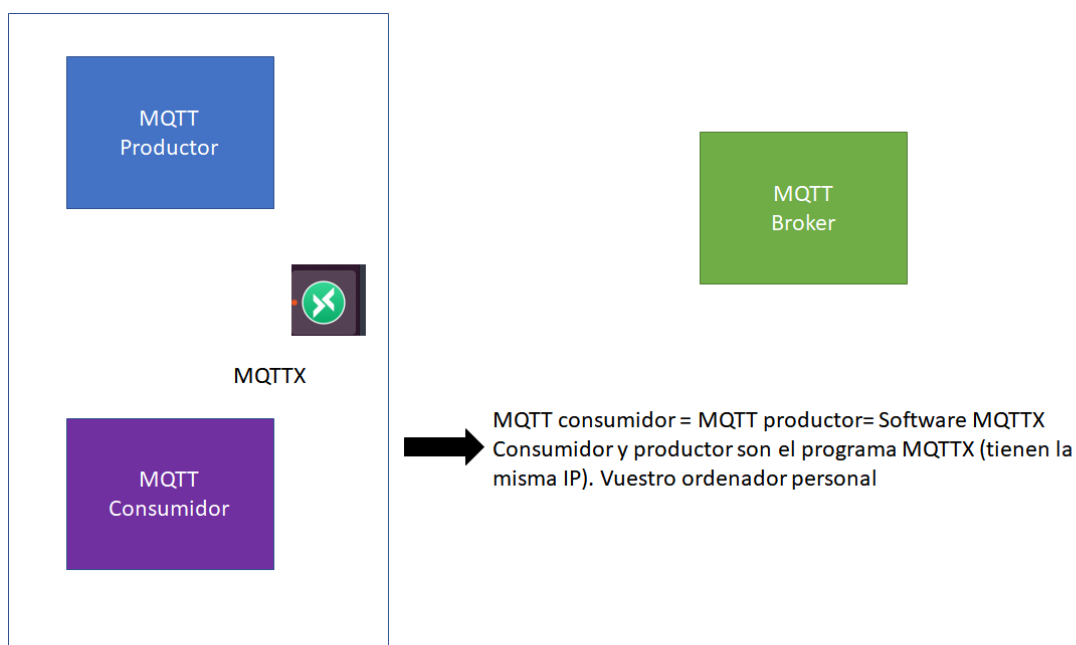
Si lo instaláis usando la interfaz gráfica se mostrará una pantalla como esta



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)



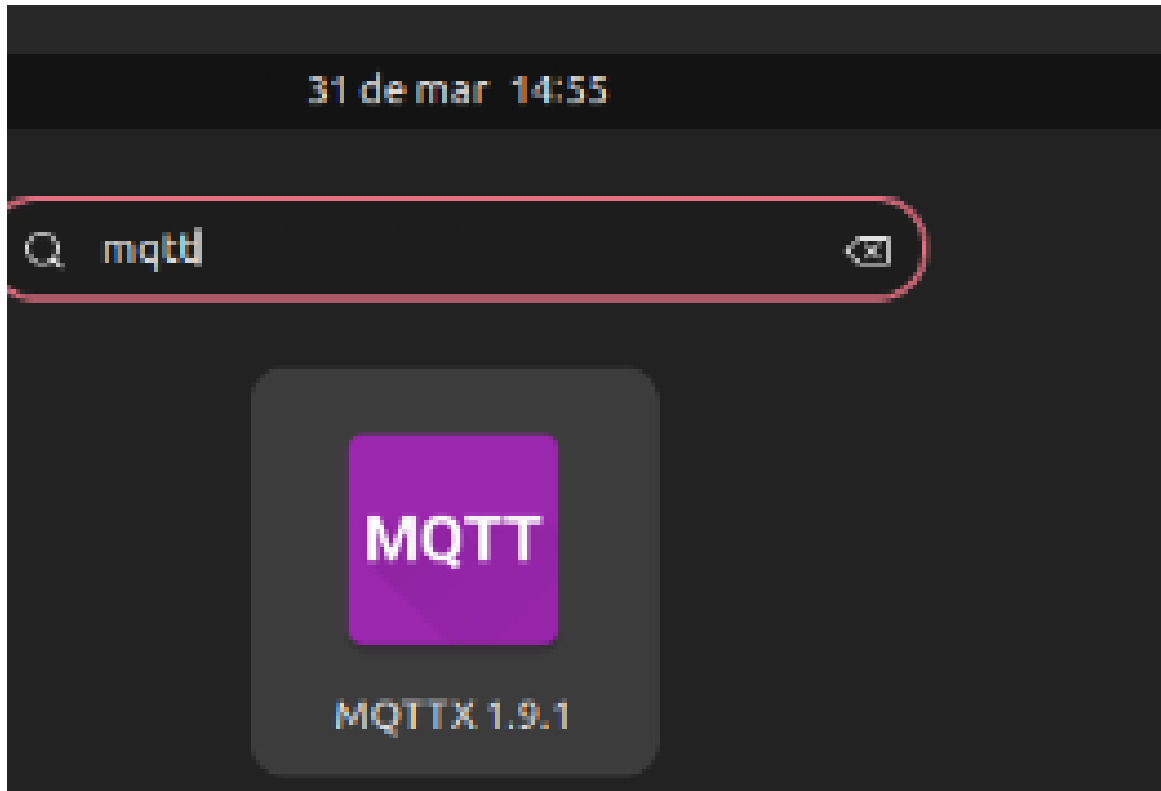
Continúa en el paso 4

### 3. Usuarios con vncweb o usuarios en los laboratorios

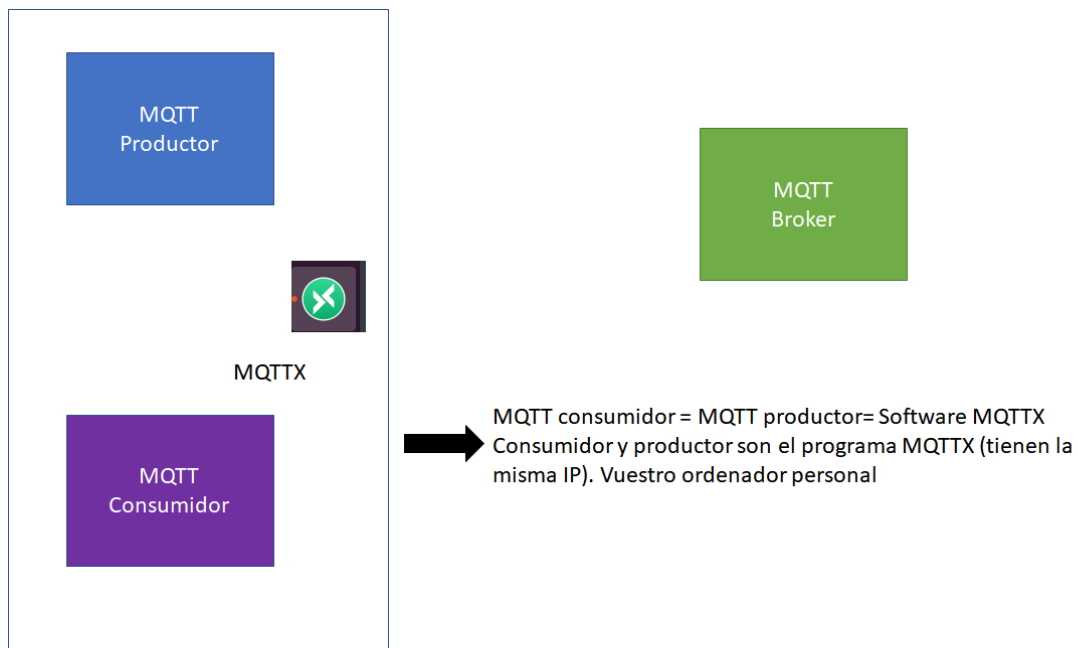
Conectate con tu usuario al laboratorio <https://labs.eif.urjc.es/vnc/> o en el pc de los laboratorios

#### 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

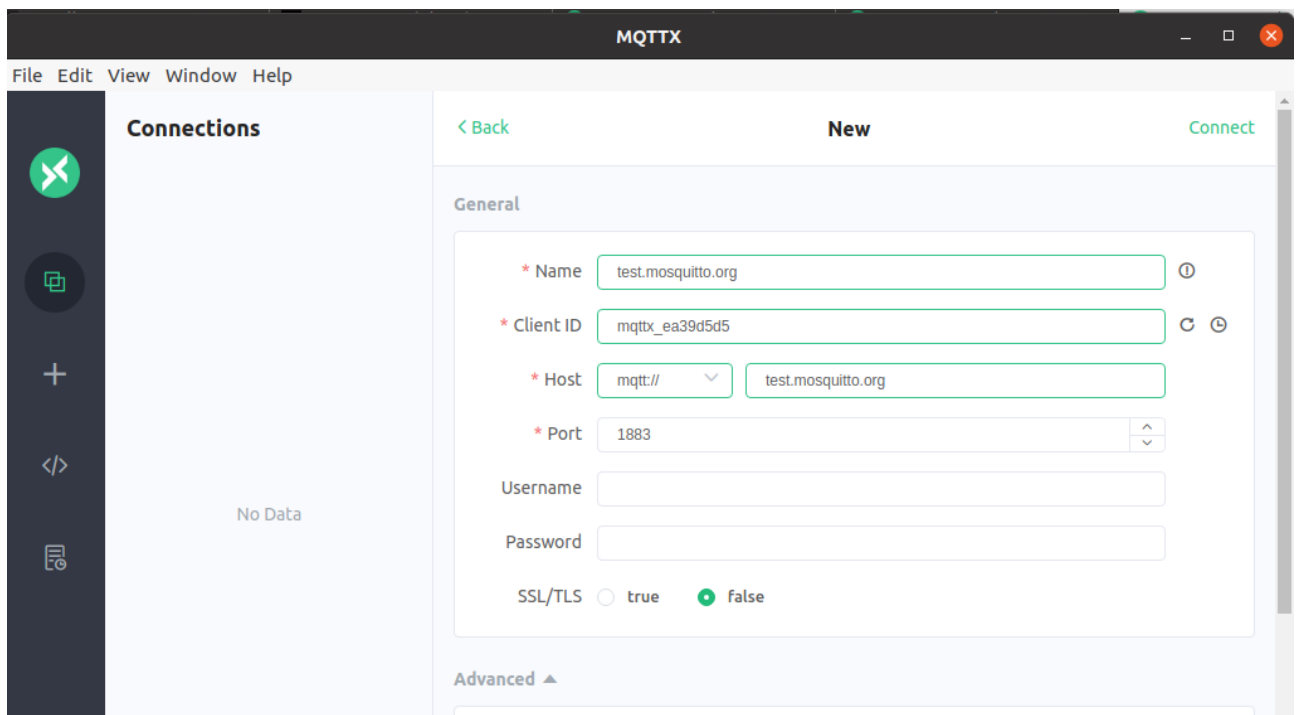


Con las siguientes propiedades

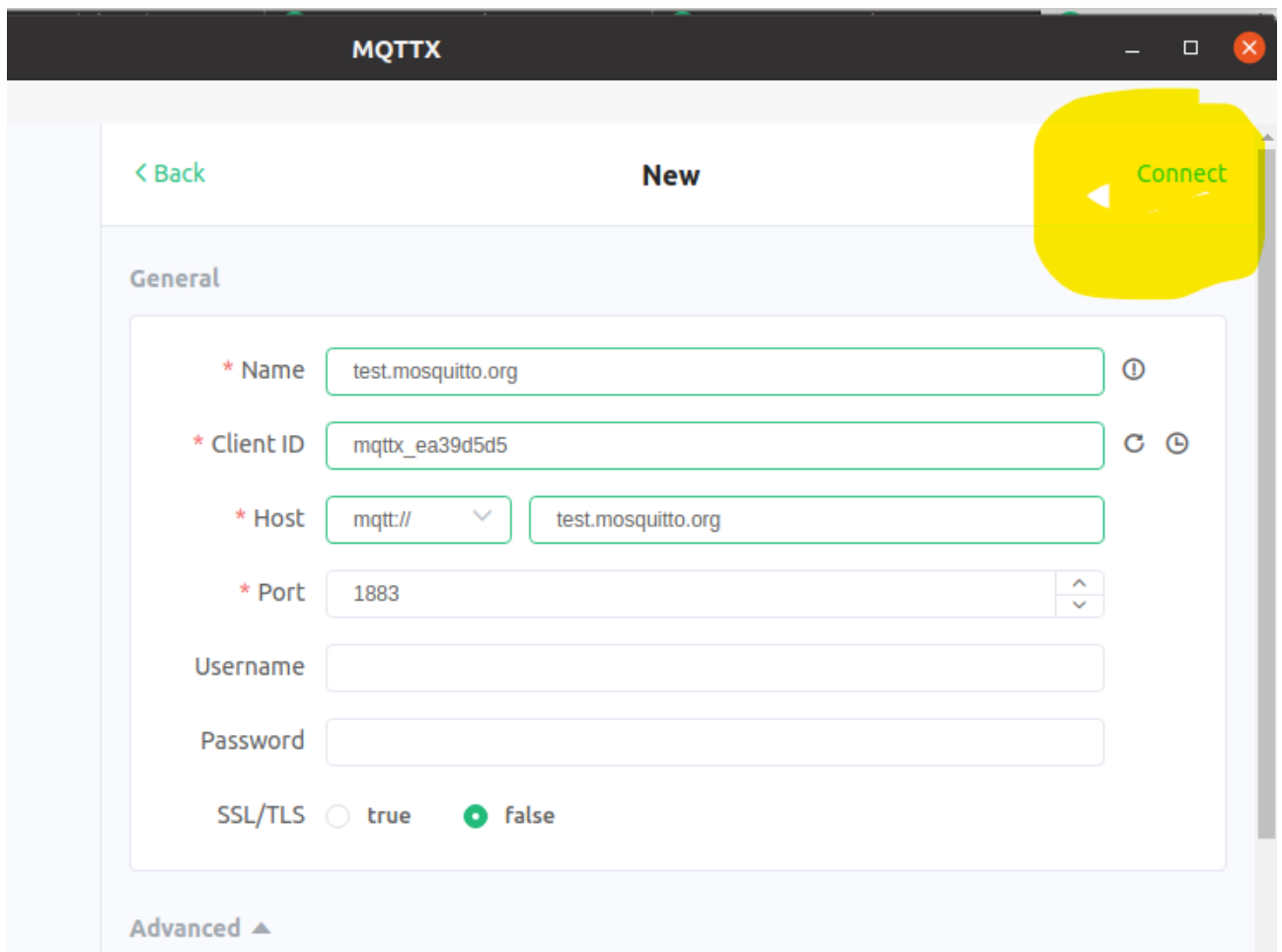
Name: test.mosquitto.org

Host: test.mosquitto.org

Port:1883



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

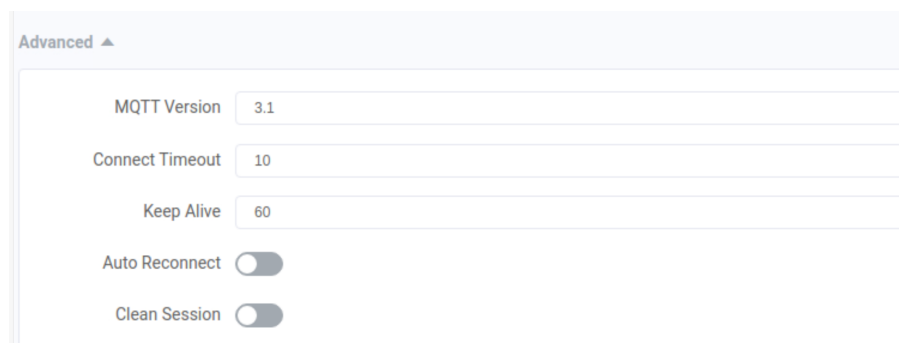


The image shows the 'New' connection configuration window in the MQTTX application. The window has a dark header with the title 'MQTTX'. Below the header, there are navigation links '< Back' and 'New'. A yellow callout bubble highlights a 'Connect' button in the top right corner. The main section is titled 'General' and contains several input fields:
 

- \* Name: test.mosquitto.org
- \* Client ID: mqttx\_ea39d5d5
- \* Host: mqtt:// (dropdown) and test.mosquitto.org (text input)
- \* Port: 1883
- Username: (empty text input)
- Password: (empty text input)
- SSL/TLS: Radio buttons for 'true' and 'false', with 'false' selected.

 At the bottom, there is an 'Advanced' section with a small upward arrow icon.

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



The image shows the 'Advanced' configuration section of the MQTTX application. It contains several settings:
 

- MQTT Version: 3.1
- Connect Timeout: 10
- Keep Alive: 60
- Auto Reconnect: Toggle switch (disabled)
- Clean Session: Toggle switch (disabled)

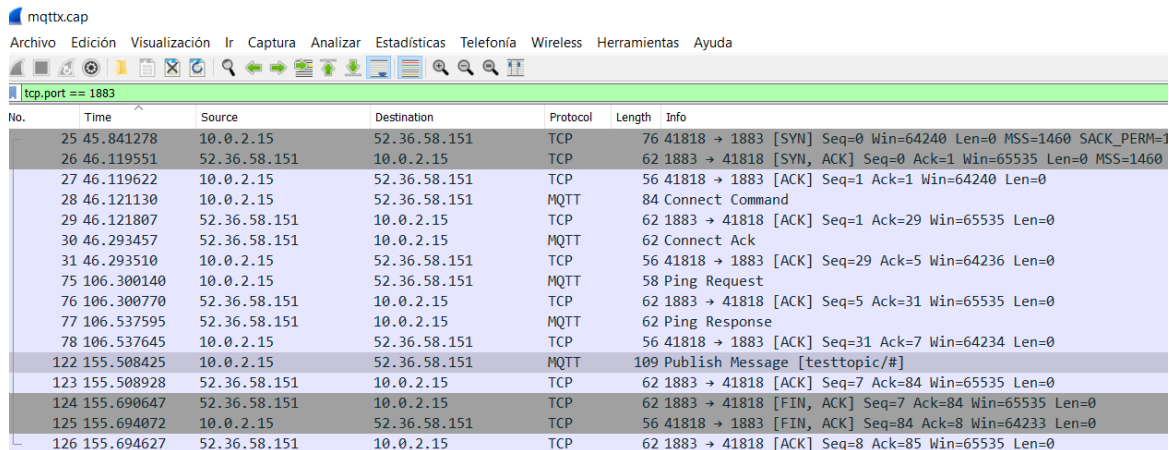
La conexión se pondrá en verde:



The image shows the 'Connections' panel in the MQTTX application. It displays a list of connections. The first connection is 'test.mosquitto.org' with a green checkmark and a '0' in a circle. Below it, there is a '+ New Subscription' button and a 'Plaintext' dropdown menu. The connection 'test.mosquitto.org@test...' is highlighted with a green background.

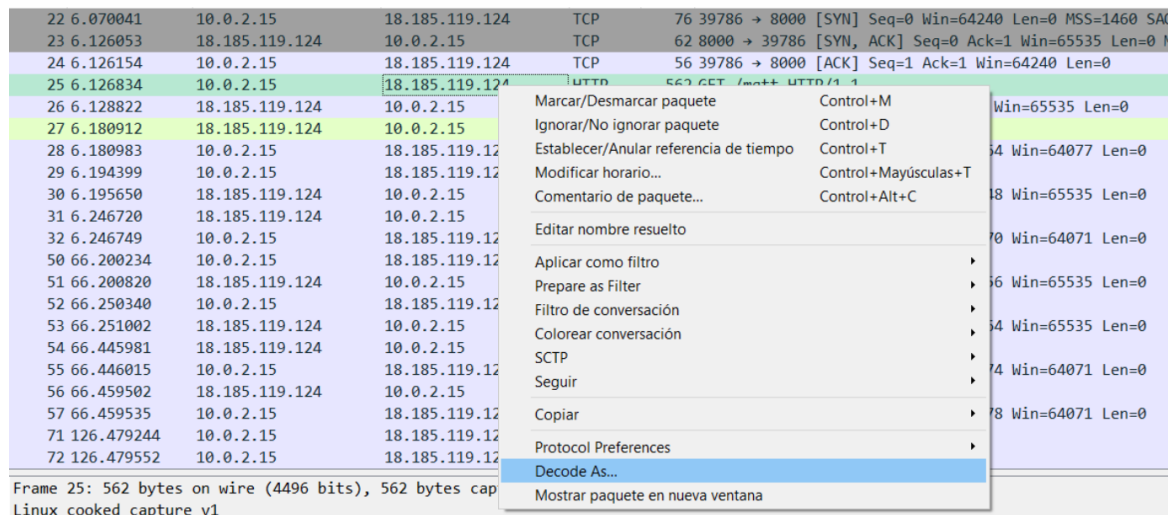
4. Para la captura con Control+C.
5. Abre la captura en wireshark. Si no tienes permisos para abrir la captura, cámbiaselos con `sudo chmod`.

La conexión de este MQTT cliente se realiza utilizando el puerto 1883. En algunas versiones de wireshark 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`



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...**:

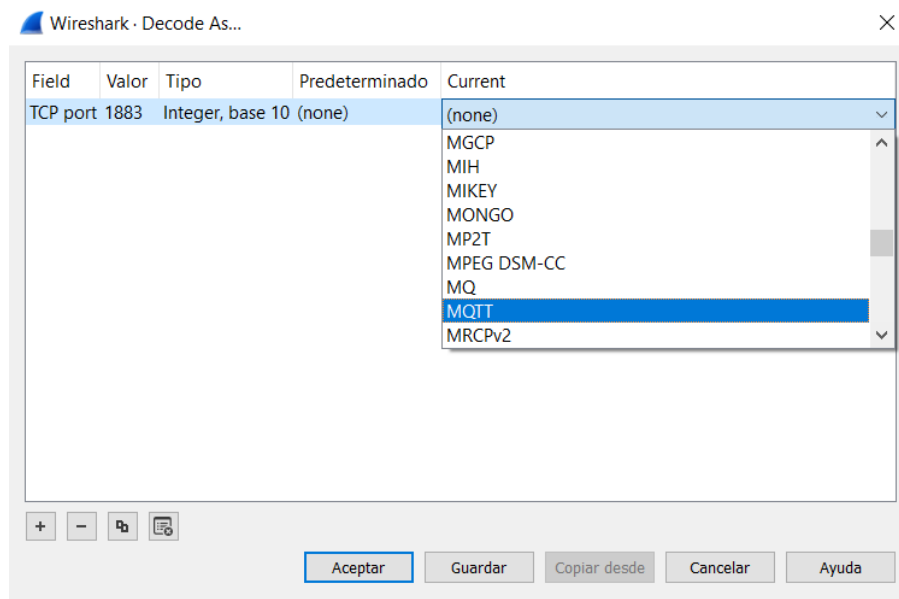


No.	Time	Source	Destination	Protocol	Length	Info
22	6.070041	10.0.2.15	18.185.119.124	TCP	76	39786 → 8000 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SA
23	6.126053	18.185.119.124	10.0.2.15	TCP	62	8000 → 39786 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
24	6.126154	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64240 Len=0
25	6.126834	10.0.2.15	18.185.119.124	MQTT	562	GET /mqtt/UTTO/1
26	6.128822	18.185.119.124	10.0.2.15	TCP	62	8000 → 39786 [ACK] Seq=1 Ack=1 Win=65535 Len=0
27	6.180912	18.185.119.124	10.0.2.15	TCP	62	8000 → 39786 [ACK] Seq=1 Ack=1 Win=65535 Len=0
28	6.180983	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64077 Len=0
29	6.194399	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
30	6.195650	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
31	6.246720	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
32	6.246749	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
50	66.200234	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
51	66.200820	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
52	66.250340	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
53	66.251002	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
54	66.445981	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
55	66.446015	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
56	66.459502	18.185.119.124	10.0.2.15	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
57	66.459535	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
71	126.479244	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0
72	126.479552	10.0.2.15	18.185.119.124	TCP	56	39786 → 8000 [ACK] Seq=1 Ack=1 Win=64071 Len=0

Frame 25: 562 bytes on wire (4496 bits), 562 bytes captured on interface v1  
Linux cooked capture v1



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 **flitro** con el nombre del protocolo: **mqtt**

**Identifica** el mensaje **Connect**.

6. ¿Qué dirección IP utiliza el cliente MQTT?
7. ¿Qué dirección IP utiliza el broker?
8. ¿Qué versión se utiliza del protocolo MQTT?
9. ¿Qué client id utiliza?
10. En el mensaje de Connection Ack, ¿qué código devuelve? ¿qué significa ese código?
11. ¿Tiene el parámetro Clean Session activo?

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

## 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
```

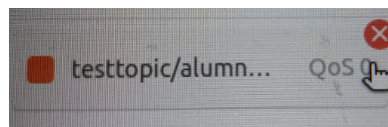
5. Subscribe tu cliente (consumidor) al siguiente tema con **Qos 2**. Asegúrate de ajustar el campo QoS de la subscripción.

```
testtopic/p7/X/Qos2
```

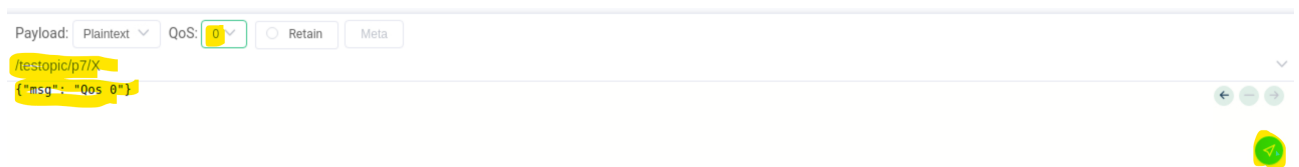
6. Para la captura con Control+C.
7. Abre la captura en wireshark. Identifica los mensajes **Subscribe**
8. ¿A qué topics te has suscrito?
9. Identifica los diferentes mensaje de subscripción por tu cliente como Qos=0, Qos=1, Qos=2 ¿cómo los has reconocido?  
Chequéalo identificando el contenido del mensaje
10. ¿Quién es el originante (productor del mensaje)? ¿Cliente o Broker?
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?

## 7. Publish

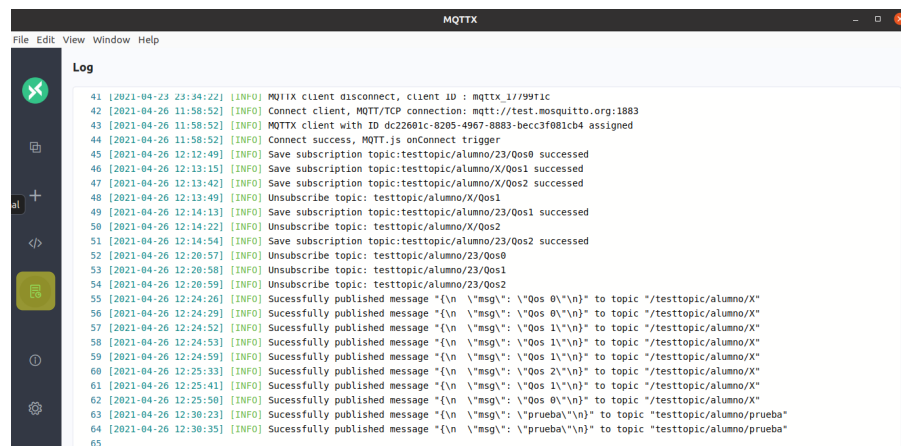
1. Elimina las subscripciones del apartado anterior:



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:



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 Identifica los mensajes **Publish**
9. 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
10. 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.
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?

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

## 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?
11. Identifica los mensajes Publish enviados por el cliente y los enviados por el broker. ¿Cómo los has reconocido?
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?
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?
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? ¿Wué mensajes de tipo Publish se transmiten desde el broker a tu cliente (consumidor)?

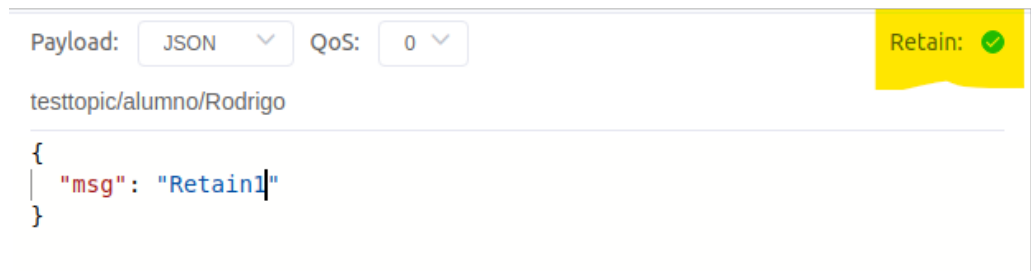
Compara el comportamiento de la subscripción con Qos=0 con el comportamiento de la subscripción con Qos=2:

15. Para la subscripción de tipo Qos=0, ¿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))?
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))?

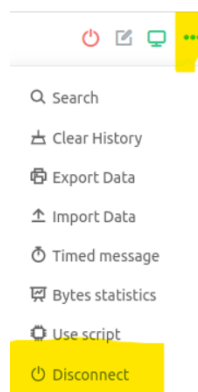
## 9. Retain

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
11. ¿qué longitud tiene el mensaje?
12. Identifica los mensajes que tienen el flag Retain en su cabecera Retain. ¿Qué mensajes son?
13. ¿Qué texto de mensaje se recibe en el consumidor una vez que se vuelves a suscribir al topic `testtopic/p7`?
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

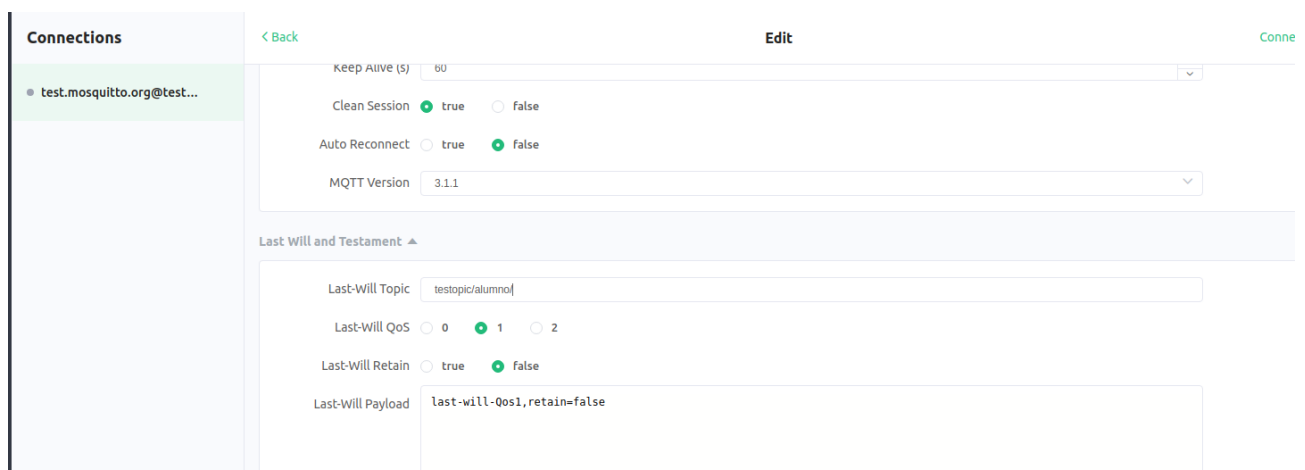
17. Desconecta la conexión.
18. Conéctate de nuevo
19. Suscríbete al mismo topic de antes pero con Qos=2.
20. Para la captura con Control+C.
21. Abre la captura en wireshark.
22. ¿Cuántos mensajes Publish Message se envían con el flag retain activo? ¿Son los mensajes publicados por la misma IP? ¿Puedes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

## 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
6. ¿De qué topic estás quitando la subscripción?

## 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.



The screenshot shows the 'Connections' tab in an MQTT client interface. A connection to 'test.mosquitto.org@test...' is selected. The 'Edit' panel for this connection shows the following settings:

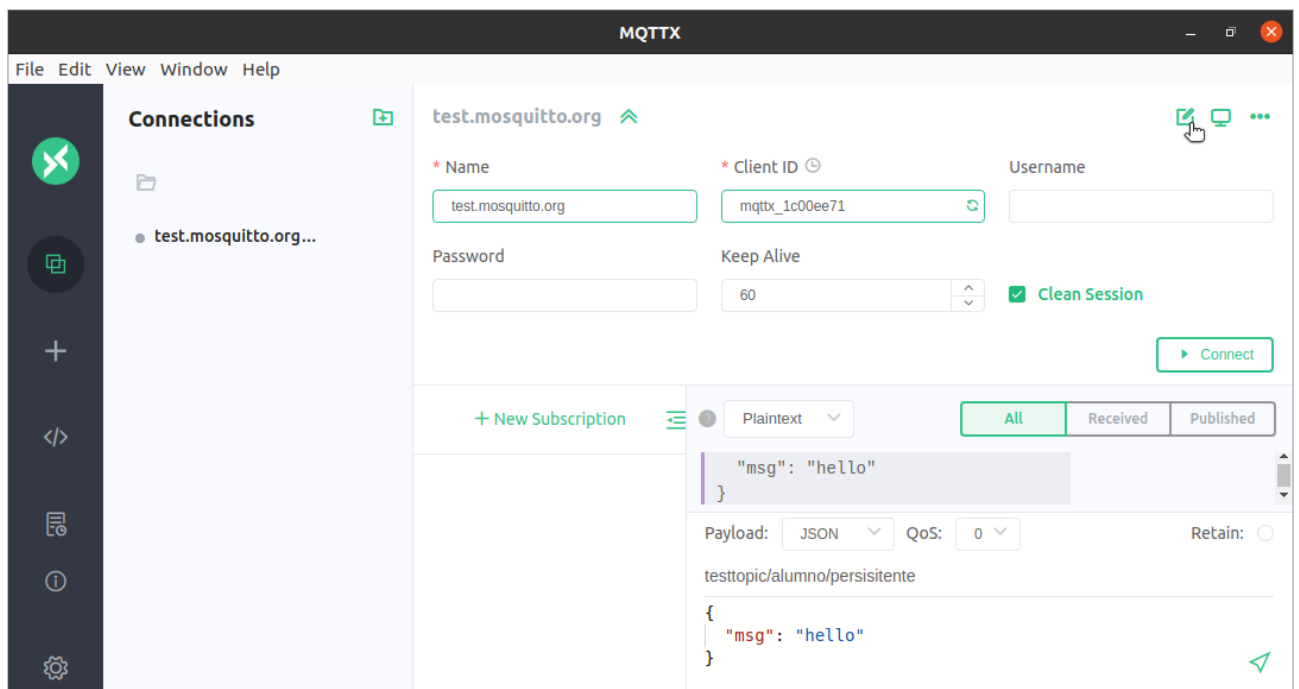
- Keep Alive (s): 60
- Clean Session: ☒ true ☐ false
- Auto Reconnect: ☐ true ☒ false
- MQTT Version: 3.1.1
- Last Will and Testament** (expanded):
  - Last-Will Topic: testtopic/alumno01
  - Last-Will QoS: ☐ 0 ☒ 1 ☐ 2
  - Last-Will Retain: ☐ true ☒ false
  - Last-Will Payload: last-will-Qos1,retain=false

3. Los parámetros son:
    - LastWillTopic=testtopic/p7/
    - LastWill-Qos=1
    - LastWill-Retain=False
    - LastWill Message
- Incluid el siguiente mensaje: LastWillQos1,RetainFalse

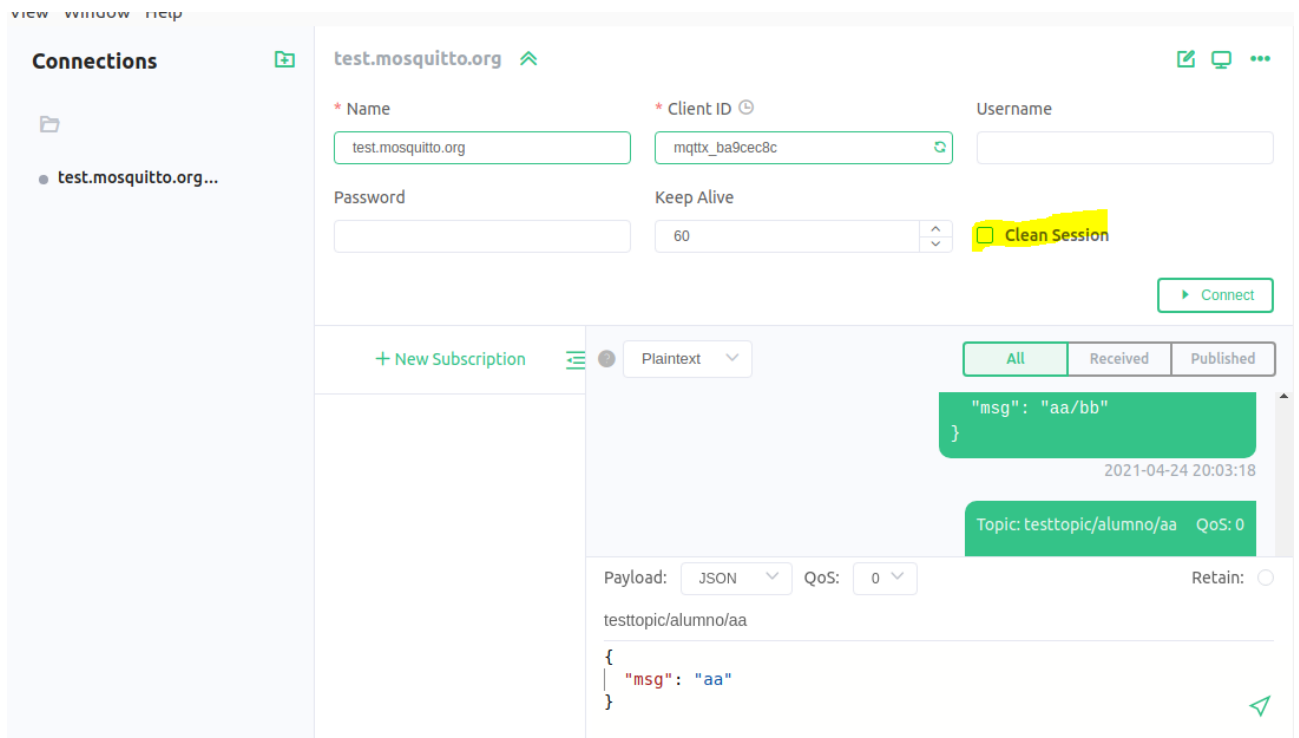
4. Verifica que la conexión está activa.
5. Elimina cualquier subscripción que tengas
6. Crea una subscripción `testtopic/p7/X/+`
7. Publica el siguiente topic: `testtopic/p7/X/aa/bb`
8. Publica el siguiente topic: `testtopic/p7/X/aa`
9. ¿Cuántos mensajes se reciben por parte del consumidor? ¿Por qué?
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?
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?

## 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?

## Normas de entrega

Es necesario entregar la siguiente documentación:

- Memoria en formato pdf donde se responda razonadamente a las cuestiones planteadas en este enunciado.
- Fichero p7.tgz que incluya las capturas de tráfico: desde mqtt-01.cap hasta mqtt-11.cap.

## 13. Anexo A: Captura de tráfico

### 13.1. Captura de tráfico con comandos en tu ordenador

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

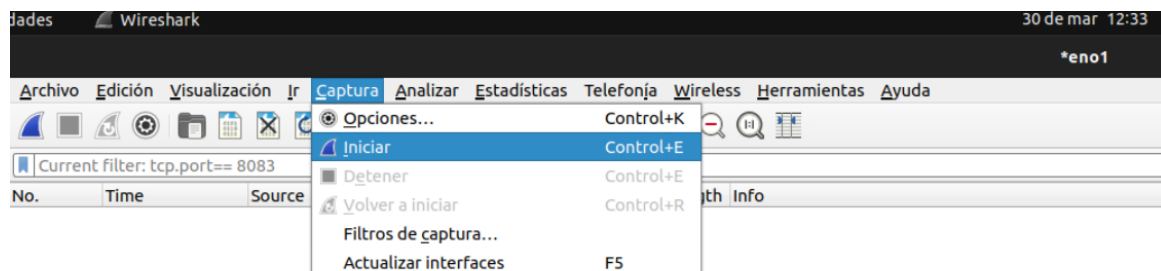
```
sudo tcpdump -i any -s 0 -w <nombre captura>
```

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**.

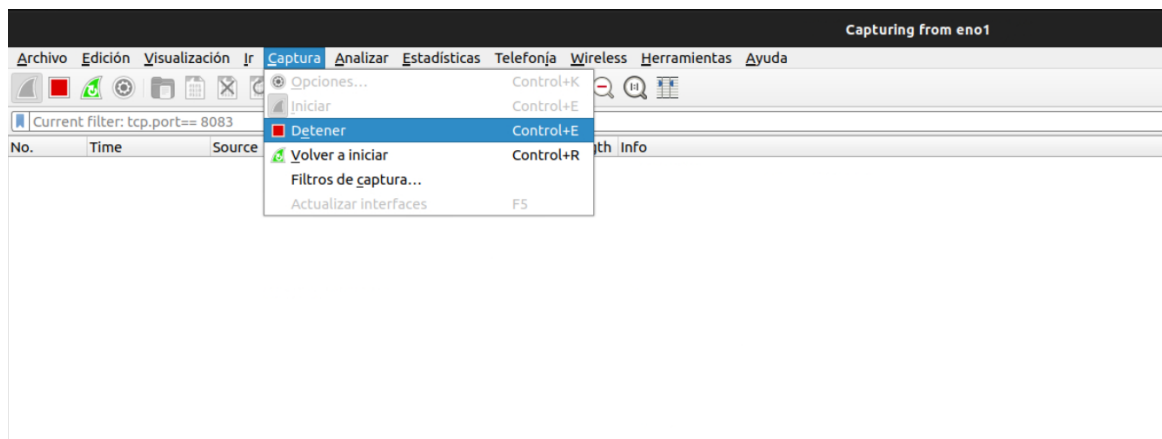
Para parar la captura usa Ctrl+C

### 13.2. Captura de tráfico usando vnc

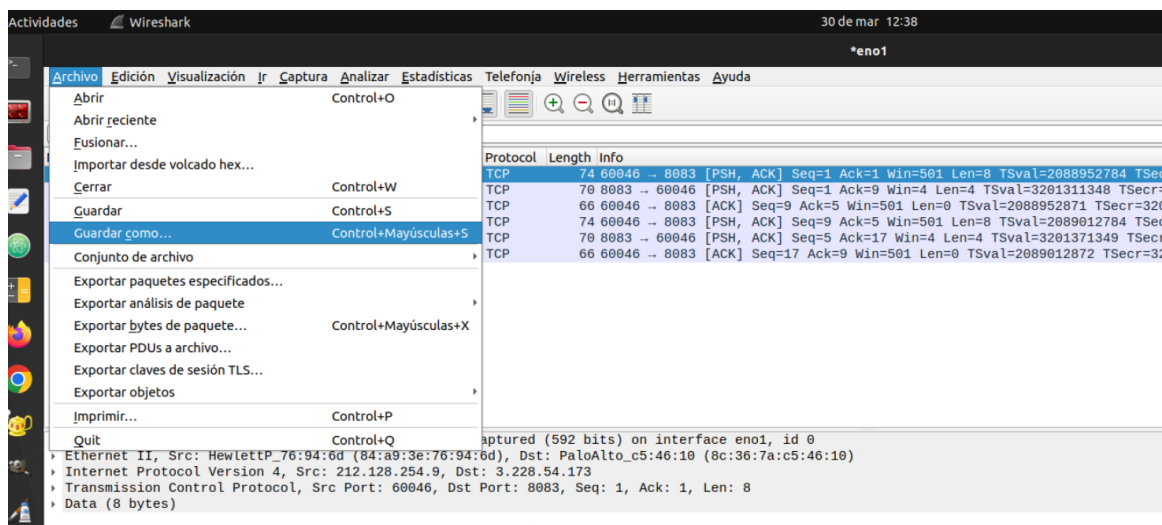
Abre la aplicación wireshark Captura tráfico tal y como se indica en la figura



Para detener la captura



Para guardar la captura



## 14. Anexo B: Servidor alternativo de mqttx

En el caso que el servidor definido en la parte Connect paso 4 no se encuentre disponible, hay un servidor mqtt alternativo que se puede acceder **sólo para los usuarios que se conecten usando la conexion vnc** (ver paso) 3

Sus propiedades son:

Name: lbdd.aulas.eif.urjc.es

Host: lbdd.aulas.eif.urjc.es

Port:1883

## General

\* Name

\* Client ID

\* Host

\* Port

Username

Password

SSL/TLS ☐

## Advanced ▲

MQTT Version

Connect Timeout

Keep Alive