

Tarea 1: Introducción a MQTT

El objetivo de este ejercicio es familiarizarse de forma práctica con la securización básica de una aplicación IoT basada en un broker MQTT.

Se recomienda usar como broker de sandbox: <https://www.emqx.io/mqtt/public-mqtt5-broker>

Ejercicio #1: Conexión y publicación en broker MQTT conn MQTT.fx (puntuación 20%)

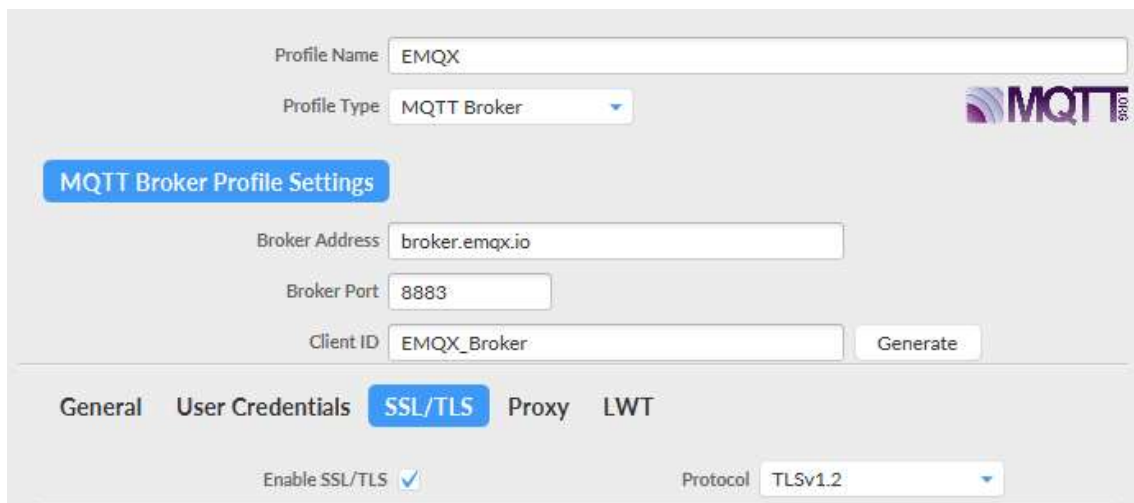
Para ello debéis usar en primer lugar el cliente gráfico:

<http://mqttfx.jensd.de/index.php/download>

Resultado: capturas de pantalla de settings de conexión y publicación usando MQTT.FX así como captura de pantalla de la línea de comandos publicando con mosquitto-client.

MQTT.FX:

Configuración:



Publicación:



Publicando con Mosquitto:

```
mosquitto_pub -t mcs/01_Iker -m "hola desde mosquitto" -h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
```

```
C:\Users\Admin\Documents\Tech\ENIIT\Ciberseguridad\09_Seguridad_IoT\01_Tarea>mosquitto_pub -t mcs/01_Iker -m "hola desde mosquitto"
-h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q0, r0, m1, 'mcs/01_Iker', ... (20 bytes))
Client null sending DISCONNECT
```

Subscripción:

mcs/01_lker

Subscribe

QoS 0QoS 1QoS 2Autoscroll

mcs/01_lker

4

Dump MessagesMuteUnsubscribe

Topics Collector (0)

ScanStop

mcs/01_lker

1

QoS 0

mcs/01_lker

2

QoS 0

mcs/01_lker

3

QoS 0

mcs/01_lker

4

QoS 0

mcs/01_lker

4

16-01-2022 13:40:44.49244711

QoS 0

Muy bien. Estoy haciendo los ejercicios de IoT

mcs/01_lker

Subscribe

QoS 0QoS 1QoS 2Autoscroll

mcs/01_lker

5

Dump MessagesMuteUnsubscribe

Topics Collector (0)

ScanStop

mcs/01_lker

1

QoS 0

mcs/01_lker

2

QoS 0

mcs/01_lker

3

QoS 0

mcs/01_lker

4

QoS 0

mcs/01_lker

5

QoS 0

mcs/01_lker

5

16-01-2022 13:42:52.49372529

QoS 0

hola desde mosquitto

Ejercicio #2: Conexión y publicación en broker MQTT con Mosquitto (puntuación 20%)

Para ello debéis usar en primer lugar el cliente mosquitto que forma parte de las utilidades de broker:

<https://mosquitto.org/download>

Resultado: captura de pantalla de publicación y la recepción del mensaje mediante una suscripción al mismo topic con mosquitto_pub y mosquitto_sub

Publicación:

```
mosquitto_pub -t mcs/handballsensors -m "{\"id\":\"11\",\"hand\":\"left\",\"speed\":112,\"goal\":1}" -h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
```

```
C:\Users\Admin\Documents\Tech\ENIIT\Ciberseguridad\09_Seguridad_IoT\01_Tarea>mosquitto_pub -t mcs/handballsensors -m "{\"id\":\"11\",\"hand\":\"left\",\"speed\":112,\"goal\":1}" -h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending PUBLISH (d0, q0, r0, m1, 'mcs/handballsensors', ... (44 bytes))
Client null sending DISCONNECT
```

Subscripción:

```
mosquitto_sub -t mcs/handballsensors -h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
```

```
C:\Users\Admin\Documents\Tech\ENIIT\Ciberseguridad\09_Seguridad_IoT\01_Tarea>mosquitto_sub -t mcs/handballsensors -h broker.emqx.io -p 8883 --cafile ./broker.emqx.io-ca.crt --debug
Client null sending CONNECT
Client null received CONNACK (0)
Client null sending SUBSCRIBE (Mid: 1, Topic: mcs/handballsensors, QoS: 0, Options: 0x00)
Client null received SUBACK
Subscribed (mid: 1): 0
Client null received PUBLISH (d0, q0, r0, m0, 'mcs/handballsensors', ... (44 bytes))
{"id":11,"hand":"left","speed":112,"goal":1}
Client null sending PINGREQ
Client null received PINGRESP
```

Ejercicio #3: Dashboard (puntuación 60%)

Una vez que hayas publicado, vamos a realizar un dashboard sencillo con: <https://freeboard.io>

Resultado: captura de pantalla de publicación y URL del dashboard, se solicita usar varios widgets diferentes para ilustrar un caso complejo.

<https://freeboard.io/board/INg-yV>

