

Práctica 5 IoT

3. Connect

Time	Source	Destination	Protocol	Length	Info
15 0.631655	192.168.0.169	91.121.93.94	MQTT	102	Connect Command

Frame 15: 102 bytes on wire (816 bits), 102 bytes captured (816 bits)
Linux cooked capture v2
Internet Protocol Version 4, Src: 192.168.0.169, Dst: 91.121.93.94
Transmission Control Protocol, Src Port: 42552, Dst Port: 1883, Seq: 1, Ack: 1, Len: 30
MQ Telemetry Transport Protocol, Connect Command
▸ Header Flags: 0x10, Message Type: Connect Command
Msg Len: 28
Protocol Name Length: 4
Protocol Name: MQTT
Version: MQTT v3.1.1 (4)
▸ Connect Flags: 0x82, User Name Flag, QoS Level: At most once delivery (Fire and Forget), Clean Session Flag
Keep Alive: 60
Client ID Length: 14
Client ID: mqttx_d5bd83e8
User Name Length: 0
User Name:

3.7 La dirección IP que utiliza el *cliente* MQTT es **192.168.0.169**.

3.8 La dirección IP que utiliza el *broker* es **91.121.93.94**.

3.9 La versión que utiliza el protocolo MQTT es la **versión 3.1.1**.

3.10 El *client id* que utiliza es **mqttx_d5bd83e8**.

3.11

Time	Source	Destination	Protocol	Length	Info
17 0.663099	91.121.93.94	192.168.0.169	MQTT	76	Connect Ack

Frame 17: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
Linux cooked capture v2
Internet Protocol Version 4, Src: 91.121.93.94, Dst: 192.168.0.169
Transmission Control Protocol, Src Port: 1883, Dst Port: 42552, Seq: 1, Ack: 31, Len: 4
MQ Telemetry Transport Protocol, Connect Ack
▾ Header Flags: 0x20, Message Type: Connect Ack
0010 = Message Type: Connect Ack (2)
.... 0000 = Reserved: 0
Msg Len: 2
▾ Acknowledge Flags: 0x00
0000 000. = Reserved: Not set
.... ...0 = Session Present: Not set
Return Code: Connection Accepted (0)

El código que devuelve es **Connection Accepted**. Este código quiere decir que la conexión se realizó con éxito, el broker aceptó la conexión.

3.12 No tiene el parámetro **Clean Flag** ya que dicho parámetro lo emplea la versión 5 de MQTT.

4. Ping

4.5 Los mensajes **Ping** se producen cada 60 segundos. Dicho valor se encuentra configurado en el *keep alive* en el connect.

4.6 La longitud del mensaje que tiene el *ping request* es 2 bytes.

4.7 La longitud del mensaje que tiene el *ping response* es 2 bytes.

5. Subscribe

No.	Time	Source	Destination	Protocol	Length	Info
313	185.845368	192.168.0.169	91.121.93.94	MQTT	100	Subscribe Request (id=456) [testtopic/p5/188/Qos0]
315	186.113401	91.121.93.94	192.168.0.169	MQTT	77	Subscribe Ack (id=456)
320	201.417580	192.168.0.169	91.121.93.94	MQTT	100	Subscribe Request (id=457) [testtopic/p5/188/Qos1]
322	201.455608	91.121.93.94	192.168.0.169	MQTT	77	Subscribe Ack (id=457)
346	217.557245	192.168.0.169	91.121.93.94	MQTT	100	Subscribe Request (id=458) [testtopic/p5/188/Qos2]
347	217.590643	91.121.93.94	192.168.0.169	MQTT	77	Subscribe Ack (id=458)

5.8 Los topics a los que me he suscrito son *testtopic/p5/188/Qos0*, *testtopic/p5/188/Qos1* y *testtopic/p5/188/Qos2*.

5.9 Se reconocen los diferentes mensajes de suscripción por el campo “**Requested QoS**”, el cual indica el tipo de calidad de servicio con el que se está suscribiendo.

5.10 El productor del mensaje es el cliente.

5.11 El campo que utiliza para identificar el topic del Subscribe Request en cada uno de los mensajes es “**Message Identifier**”. Hay 3 mensajes *Subscribe* y 3 mensajes *Subscribe Ack*.

6. Publish

	Time	Source	Destination	Protocol	Length	Info
97	63.410199	192.168.0.169	91.121.93.94	MQTT	112	Publish Message [testtopic/p5/188]
126	90.662693	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=42221) [testtopic/p5/188]
128	91.034740	91.121.93.94	192.168.0.169	MQTT	76	Publish Ack (id=42221)
130	96.224446	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=42222) [testtopic/p5/188]
132	96.273016	91.121.93.94	192.168.0.169	MQTT	76	Publish Received (id=42222)
134	96.274932	192.168.0.169	91.121.93.94	MQTT	76	Publish Release (id=42222)
135	96.311077	91.121.93.94	192.168.0.169	MQTT	76	Publish Complete (id=42222)

6.9 El mensaje publicado por el cliente como Qos = 0 es el paquete 97. Se identifica gracias al campo “**QoS Level**”, el cual indica la calidad de servicio con el que se ha publicado el mensaje. Dicho campo se encuentra en los flags de la cabecera. No hay ningún mensaje *Publish Ack* a dicho mensaje Publish.

6.10 El mensaje publicado por el cliente como Qos = 1 es el paquete 126. Al igual que en el apartado anterior se identifica gracias al campo “**QoS Level**”. Sí, hay un mensaje *Publish Ack* para ese mensaje Publish. La asociación entre el *Publish Request* y el *Publish Ack* se realiza mediante el campo “**Message Identifier**” que poseen ambos mensajes y permiten identificar el Publish al que se quiere referir.

6.11 El mensaje publicado por el cliente como Qos = 2 es el paquete 130. Al igual que en el apartado 6.9 se identifica gracias al campo “**QoS Level**”. No hay un mensaje *Publish Ack* para ese mensaje Publish. Hay otros mensajes como *Publish Received*, *Publish Release* y *Publish Complete*. La asociación entre el *Publish Release* y el *Publish Complete* es la misma que con los mensajes *Publish Request* y el *Publish Ack*, se realiza mediante el campo “**Message Identifier**” que poseen ambos mensajes.

7. Subscribe-Publish-Qos

7.1 Suscripción con Qos=0

	Time	Source	Destination	Protocol	Length	Info
35	24.800187	192.168.0.169	91.121.93.94	MQTT	97	Subscribe Request (id=42226) [testtopic/p5/188/#]
36	24.833089	91.121.93.94	192.168.0.169	MQTT	77	Subscribe Ack (id=42226)
54	37.088416	192.168.0.169	91.121.93.94	MQTT	112	Publish Message [testtopic/p5/188]
55	37.121729	91.121.93.94	192.168.0.169	MQTT	112	Publish Message [testtopic/p5/188]
58	47.097891	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=42227) [testtopic/p5/188]
59	47.131409	91.121.93.94	192.168.0.169	MQTT	112	Publish Message [testtopic/p5/188]
61	47.135720	91.121.93.94	192.168.0.169	MQTT	76	Publish Ack (id=42227)
65	51.970005	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=42228) [testtopic/p5/188]
66	52.003964	91.121.93.94	192.168.0.169	MQTT	76	Publish Received (id=42228)
68	52.004724	192.168.0.169	91.121.93.94	MQTT	76	Publish Release (id=42228)
69	52.035302	91.121.93.94	192.168.0.169	MQTT	112	Publish Message [testtopic/p5/188]
70	52.039467	91.121.93.94	192.168.0.169	MQTT	76	Publish Complete (id=42228)

7.1.9 Se identifica que la suscripción hecha tiene un Qos=0 por el campo “**Requested QoS**” y el “**Reason Code**” que nos garantiza que la suscripción se realizó con éxito.

7.1.10 Los mensajes *Publish* enviados por el cliente son los paquetes 54, 58 y 65. Los mensajes *Publish* enviados por el broker son los paquetes 55, 59 y 69. Se reconocen por las direcciones IP origen y destino.

7.1.11 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=0 es el paquete 54. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=0 es el paquete 55.

7.1.12 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=1 es el paquete 58. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=1 es el paquete 59 y 61. Solo hay un mensaje *Publish Ack*. El mensaje de *Publish* desde el broker al consumidor se realiza antes del mensaje *Publish Ack*.

7.1.13 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=2 es el paquete 65 y 68. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=2 es el paquete 66, 69 y 70. No hay ningún mensaje *Publish Ack*. El mensaje de *Publish* desde el broker al consumidor se realiza después de los mensajes *Publish Received* y *Publish Release*.

7.2 Suscripción con Qos=2

Time	Source	Destination	Protocol	Length	Info
85 56.564278	192.168.0.169	91.121.93.94	MQTT	112	Publish Message [testtopic/p5/188]
86 56.611955	91.121.93.94	192.168.0.169	MQTT	112	Publish Message [testtopic/p5/188]
91 65.027738	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=8699) [testtopic/p5/188]
92 65.059291	91.121.93.94	192.168.0.169	MQTT	114	Publish Message (id=1) [testtopic/p5/188]
94 65.064457	91.121.93.94	192.168.0.169	MQTT	76	Publish Ack (id=8699)
96 65.080881	192.168.0.169	91.121.93.94	MQTT	76	Publish Ack (id=1)
100 70.924047	192.168.0.169	91.121.93.94	MQTT	114	Publish Message (id=8700) [testtopic/p5/188]
102 70.961510	91.121.93.94	192.168.0.169	MQTT	76	Publish Received (id=8700)
103 70.962771	192.168.0.169	91.121.93.94	MQTT	76	Publish Release (id=8700)
104 70.993039	91.121.93.94	192.168.0.169	MQTT	114	Publish Message (id=2) [testtopic/p5/188]
105 70.994696	192.168.0.169	91.121.93.94	MQTT	76	Publish Received (id=2)
106 70.997044	91.121.93.94	192.168.0.169	MQTT	76	Publish Complete (id=8700)
107 71.025270	91.121.93.94	192.168.0.169	MQTT	76	Publish Release (id=2)
109 71.047813	192.168.0.169	91.121.93.94	MQTT	76	Publish Complete (id=2)

7.2.10 La suscripción hecha tiene un Qos=2 tal como se puede observar en el campo “**Requested QoS**” del mensaje de suscripción.

7.2.11 Los mensajes *Publish* enviados por el cliente son los paquetes 85, 91 y 100. Los mensajes *Publish* enviados por el broker son los paquetes 86, 92 y 104. Se reconocen por las direcciones IP origen y destino.

7.2.12 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=0 es el paquete 85. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=0 es el paquete 86. No se observa ningún mensaje de tipo *Publish Complete* porque al publicar un mensaje con Qos=0 no se requiere confirmación de entrega a pesar de que la suscripción sea con Qos=2.

7.2.13 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=1 es el paquete 91 y 96. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=1 es el paquete 92 y 94. Hay dos mensajes *Publish Ack*, uno del cliente al broker, y otro del broker al cliente. El mensaje de *Publish* desde el broker al consumidor se realiza antes del mensaje *Publish Ack*. No se observa ningún mensaje de tipo *Publish Complete*.

7.2.14 Los mensajes de tipo *Publish* que se envían del cliente al broker con Qos=2 es el paquete 100, 103, 105 y 109. Los mensajes de tipo *Publish* que se envían del broker al cliente con Qos=2 es el paquete 102, 104, 106 y 107.

7.2.15 Para la suscripción de tipo Qos=0 la Qos desde el productor al broker para cada uno de los mensajes *Publish* generados es la misma calidad de servicio con la que envía el mensaje el productor. La Qos desde el broker al consumidor para cada uno de los mensajes *Publish* generados es 0.

la Qos productor-consumidor en cada caso desde un punto de vista global de la comunicación para cada uno de los mensajes *Publish* generados es 0, la misma Qos con la que se realizó la suscripción.

7.2.16 Para la suscripción de tipo Qos=2 la Qos desde el productor al broker para cada uno de los mensajes *Publish* generados es la misma calidad de servicio con la que envía el mensaje el productor.

La Qos desde el broker al consumidor para cada uno de los mensajes *Publish* generados es la misma calidad de servicio con la que envía el mensaje el productor.

la Qos productor-consumidor en cada caso desde un punto de vista global de la comunicación para cada uno de los mensajes *Publish* generados es 0, 1 y 2, la misma Qos con la que se publica el mensaje.

8. Retain

203	88.435754	192.168.0.169	91.121.93.94	MQTT	74 Disconnect Req
Frame 203: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)					
Linux cooked capture v2					
Internet Protocol Version 4, Src: 192.168.0.169, Dst: 91.121.93.94					
Transmission Control Protocol, Src Port: 37972, Dst Port: 1883, Seq: 129, Ack: 129, Len: 2					
MQ Telemetry Transport Protocol, Disconnect Req					
[Expert Info (Note/Protocol): Unknown version (missing the CONNECT packet?)]					
[Unknown version (missing the CONNECT packet?)]					
[Severity level: Note]					
[Group: Protocol]					
Header Flags: 0xe0, Message Type: Disconnect Req					
1110 = Message Type: Disconnect Req (14)					
.... 0000 = Reserved: 0					
Msg Len: 0					

8.11 La longitud del mensaje es 2 bytes.

8.12 Los mensajes que tienen el flag *Retain* en su cabecera *Retain* son los paquetes 107, 176 y 184.

8.13 El texto de mensaje que se recibe en el consumidor una vez que se vuelve a suscribir al mismo topic es “*Retain3*”.

Time	Source	Destination	Protocol	Length	Info
14	17.475776	192.168.0.169	91.121.93.94	MQTT	74 Ping Request
15	17.509759	91.121.93.94	192.168.0.169	MQTT	74 Ping Response
34	31.644681	192.168.0.169	91.121.93.94	MQTT	121 Publish Message (id=3178) [testtopic/p5/188]
35	31.680000	91.121.93.94	192.168.0.169	MQTT	76 Publish Received (id=3178)
37	31.680946	192.168.0.169	91.121.93.94	MQTT	76 Publish Release (id=3178)
38	31.712129	91.121.93.94	192.168.0.169	MQTT	121 Publish Message (id=1) [testtopic/p5/188]
39	31.713058	192.168.0.169	91.121.93.94	MQTT	76 Publish Received (id=1)
40	31.715213	91.121.93.94	192.168.0.169	MQTT	76 Publish Complete (id=3178)
41	31.751594	91.121.93.94	192.168.0.169	MQTT	76 Publish Release (id=1)
43	31.755245	192.168.0.169	91.121.93.94	MQTT	76 Publish Complete (id=1)
46	37.584689	192.168.0.169	91.121.93.94	MQTT	74 Disconnect Req
63	41.041553	192.168.0.169	91.121.93.94	MQTT	102 Connect Command
65	41.096224	91.121.93.94	192.168.0.169	MQTT	76 Connect Ack
67	41.166574	192.168.0.169	91.121.93.94	MQTT	97 Subscribe Request (id=57858) [testtopic/p5/188/#]
68	41.198164	91.121.93.94	192.168.0.169	MQTT	77 Subscribe Ack (id=57858)
70	41.198203	91.121.93.94	192.168.0.169	MQTT	121 Publish Message (id=1) [testtopic/p5/188]
72	41.201373	192.168.0.169	91.121.93.94	MQTT	76 Publish Received (id=1)
73	41.228972	91.121.93.94	192.168.0.169	MQTT	76 Publish Release (id=1)
74	41.230354	192.168.0.169	91.121.93.94	MQTT	76 Publish Complete (id=1)

8.22 Los paquetes 34 y 70 se envían con el flag *retain* activo. No son publicados por la misma IP. El primer mensaje lo envía el cliente y el segundo lo envía el broker.

9. Unsubscribe

Time	Source	Destination	Protocol	Length	Info
10	3.930511	192.168.0.169	91.121.93.94	MQTT	96 Unsubscribe Request (id=57859)
11	3.964904	91.121.93.94	192.168.0.169	MQTT	76 Unsubscribe Ack (id=57859)

9.6 Se quita la suscripción del topic “*testtopic/p5/188/#*”.

10. Last Will y Filtros de topic

```
Connect Flags: 0x8e, User Name Flag, QoS Level: At least once delivery (Acknowledged deliver), Will Flag, Clean Session Flag
Keep Alive: 60
Client ID Length: 14
Client ID: mqttx_bd04e01b
Will Topic Length: 13
Will Topic: testtopic/p5/
Will Message Length: 24
Will Message: 4c61737457696c6c516f73312c52657461696e46616c7365
User Name Length: 0
User Name:
```

10.9 Se recibe un solo mensaje por parte del consumidor del topic “*testtopic/p5/188/aa*”.

10.13 Se reciben tres mensajes por parte del consumidor , dos del topic “*testtopic/p5/188/aa*” y uno del “*testtopic/p5/188/aa/bb*”.

10.16 En el mensaje *connect* se puede apreciar el parámetro *Last Will*. El *lastWillQos* es *Qos = 1*.

11. Persistencia Sesión

11. Si el Clean Flag se encuentra desactivado, todas las suscripciones anteriores se mantienen al volver a reconectarse (no se recibiría ningún mensaje subscribe y ningún retain), y si está activado, se limpiaría cualquier suscripción que haya como si se empezara una conexión de cero.