



Updated: March 20, 2021 / By steve

# MQTT Retained Messages Explained

## MQTT Retained Messages

Normally if a publisher publishes a message to a

topic, and no one is subscribed to that topic the message is simply **discarded** by the broker.

However the publisher can tell the broker to keep the last message on that topic by setting the **retained message flag**.

This can be very useful, as for example, if you have sensor publishing its status only when changed e.g. Door sensor. What happens if a new subscriber subscribes to this status?

## Polls

What Protocol or Protocols are You Using for Your Project (s)

- MQTT v3.1
- MQTT v5
- MQTT-SN
- MQTT over Websockets
- http

Vote

[View Results](#)

## Newsletter

Without **retained messages** the subscriber would have to wait for the status to change before it received a message.

However with the retained message the subscriber would see the current state of the sensor.

What is important to understand is that **only one message** is retained **per topic**.

The next message published on that topic **replaces** the **last retained** message for that topic.

## MQTT Retained Messages Example Overview

In this tutorial We will use a Python script to publish and subscribe to messages.

We will then examine **how retained messages work** with various **flag** and **QOS settings**.

The important thing to note is that we will publish the messages **before** the client subscribes.

You should also note that I use **clean sessions** to avoid confusion.

Join me Up



Hi - I'm Steve and welcome to my website where you can learn how to build IOT systems using Node-Red and MQTT. If you like to support this site then you should [consider making a contribution](#).

### Search



If you don't use clean sessions then you might see messages that have **been stored** but **not retained!**

The basic process is this:

1. Publish message on a topic with **retained message flag** not set, and set
2. Subscribe to message on topic
3. Monitor messages received and analyse results

We will also look at publishing multiple messages and how to **remove or delete a retained message**.

Referring to the screen shot below:

**Example 1** – Retain message flag **not set** and the new subscriber doesn't get the message i.e no message received.

**Example 2**-Retain message **flag set** and the new subscriber gets the last message as indicated by the message received message.

**Example 3**– Retain message **flag set** and we publish several messages OFF,OFF2,OFF3. However the new subscriber only gets the last message OFF3.

## Google Ads

- [About Me](#)
- [MQTT Tools](#)
- [MQTT Quizzes](#)
- [Ask Steve for Help -Hire Me](#)
- [Networking](#)

## My Youtube Channel



- [node-red](#)
- [MQTT Brokers](#)
- [mqtt and python](#)
- [Internet](#)

```

>>> No message retained when retained flag set to false
Creating client 1 with clean session set to True
Published message= OFF ,topic=bulb1 qos= 1 retain= False
Subscribed,topic=bulb1 QOS= 1
Ending
>>> ===== RESTART =====
>>> message retained when retained flag set to True
Creating client 1 with clean session set to True
Published message= OFF ,topic=bulb1 qos= 1 retain= True
Subscribed,topic=bulb1 QOS= 1
message received OFF ←---
Ending
>>> ===== RESTART =====
>>> Only last message retained
Creating client 1 with clean session set to True
Published message= OFF ,topic=bulb1 qos= 1 retain= True
Published message= OFF2 ,topic=bulb1 qos= 1 retain= True
Published message= OFF3 ←--- topic=bulb1 qos= 1 retain= True
Subscribed,topic=bulb1 QOS= 1
message received OFF3 ←---
Ending
>>> ===== RESTART =====

```

## QOS Settings and Retained Messages and Deleting Retained Messages

The screen shot below shows 4 connection examples to demonstrate QOS effects, and then how we **delete retained messages**.

**Example 1:** We check if the QOS settings have any effect by setting the **QOS to 0** . We see that the retained message is still received and the QOS of the published message has no effect..

**Example 2:** Now we try to **delete the retained message** by setting the retained message flag to **False**. However we notice that he message is **still retained**.

**Example 3:** Now we set the message to **blank** but keep the retained message flag as **False**. Again the message is still retained.

**Example 4::** Lastly we set the message to blank and the **retain flag** to **true** which **clears** the retained message.

```

==== RESTART =====
>>> QOS has no affect on retained messages
Creating client 1 with clean session set to True
Published message= OFF ,topic=bulb1 qos= 0 retain= True
Subscribed,topic=bulb1 QOS= 0
message received OFF ← No impact
Ending
>>> ===== RESTART =====
>>> Trying to delete retained message
Creating client 1 with clean session set to True
Published message= OFF ,topic=bulb1 qos= 0 retain= False
Subscribed,topic=bulb1 QOS= 0
message received OFF ← No impact
Ending
>>> ===== RESTART =====
>>>
Creating client 1 with clean session set to True
Published message= topic=bulb1 qos= 0 retain= False
Subscribed,topic=bulb1 QOS= 0
message received OFF ← No impact
Ending
>>> ===== RESTART =====
>>>
Creating client 1 with clean session set to True
Published message= topic=bulb1 qos= 0 retain= True
Subscribed,topic=bulb1 QOS= 0
message received OFF ← Works
Ending

```

The table below is a summary showing how QOS, the clean session flag, and retained message flag affects what messages are received by a new subscriber to a topic.

## Retain Message, Clean Session and QOS Table

Clean Session Flag	Retain Flag	Subscribe QOS	Publish QOS	Published Message Always Received
True	False	0	0	No
True	False	0	1	No
True	False	1	0	No
True	False	1	1	No
False	False	0	0	No
False	False	0	1	No
False	False	1	0	No
False	False	1	1	Yes – All messages
True	True	0	0	Yes –Last Message only
True	True	0	1	Yes –Last Message only
True	True	1	0	Yes –Last Message only
True	True	1	1	Yes –Last Message only
False	True	0	0	Yes –Last Message only
False	True	0	1	Yes –Last Message only
False	True	1	0	Yes –Last Message only
False	True	1	1	Yes – All messages

**Note: QOS 1 and QOS 2 produce same result. Therefore for QOS 1 in the table read 1 or 2.**

**Note:** Some aspects of how the retain flag is handled is broker implementation dependent.

## Python Code

The code to publish with retained flag set is:

```
client.publish("bulb1","test message",qos=0,retain=True)
```

where:

topic=bulb

message =test message

QOS=0

Retained message Flag=True

## Detecting a Retained Message

When you subscribe to a topic that has a retained message that message will be delivered to your client.

The client can detect that is a retained messages by examining the **message.retain flag** in the **on message callback** as shown in the code snippet below:

```
def on_message(client, userdata, mes
    print("message received ",str(m
        "topic",message.topic,"ret
    if message.retain==1:
        print("This is a retained me
```

## Scripts

I have created a Python script that will clear the retained messages on a group of topics, and also report on the topics that have retained messages.



DOWNLOAD

## Publishing Video

Here is a video that I created that covers publishing messages and also shows the use of the retain flag.

## Publishing Messages Using the Paho Pyt...



Grateful of any feedback.

## Common Questions and Answers

**Q- How do I remove or delete a retained message?**

**A-** Publish a blank message with the **retain flag** set to **true** which **clears** the retained message.



## Q- Does the QOS of a message have any effect on retained messages?

A- **No** it doesn't.

## Q- How do I know If a message has been retained?

A- You only know when you subscribe to the topic and examine the **retained flag**.

Q- If I subscribe to a topic and receive the retained message does the broker/server then delete that message?

A- No. It is retained and will be set to new clients when they subscribe.

Q- How do I delete or clear all retained messages on Mosquitto?

A- If you're **not** using the **persistent database** then the easiest way is to stop and restart mosquitto.

Otherwise you will need to subscribe to all topics to check and clear the retained messages. You can use my [Python script](#) to do this. However this is not recommended on a busy broker.

## When to Use Retained Messages

Generally you will publish a message with the retained flag set when the message contains persistent data.

For example a sensor could publish information about itself like firmware version number, IP address, Current state.

This information is unlikely to change and so you only need to publish it once using the retain flag and any new clients can retrieve that information.

See the [Homie convention](#) for example sensor topics and attributes.

## Summary

The **retained message** feature is useful feature for keeping the last state of an object, and is especially useful when the state doesn't change frequently.

Quality of service settings don't impact **retained messages**.

## MQTT by Example Series

- [Persistent Connections- Example](#)
- [Last will and testament-Example](#)
- [MQTT Keep Alive Interval Explained With Examples](#)
- [Quiz- Test knowledge of retained messages,clean sessions and last will messages](#)

## Related tutorials and Resources: