|  |
| --- |
|  |
| Funzionamento protocollo MQTT |
|  |
| A cura dell’Ing. Buttolo Marco |

22 novembre 2017

Autore: Marco Buttolo

Funzionamento protocollo MQTT

A cura dell’Ing. Buttolo Marco

Lo scopo di questo documento è quello di descrivere sommariamente cosa è l’MQTT e mostrare come funziona.

Innanzitutto **MQTT** è l’acronimo di MQ Telemetry Transport ed è un protocollo di trasmissione di dati tra due o più dispositivi (protocolli M2M).

Possiede delle caratteristiche che rendono tale protocollo un ottimo strumento per chi vuole fare IOT, come per esempio:

* è di tipo ***publish/subscribe***
* è **molto semplice**
* ha un ***overhead*** (= quantità di dati da trasmettere per il suo funzionamento) molto basso

Si consideri il seguente esempio. Si supponga di avere a disposizione un sensore di temperatura. Si supponga di voler installare vari dispositivi nella stanza che possano accedere ai dati di temperatura letti dal sensore. Per fare ciò è sufficiente che il sensore pubblichi il dato su un **BROKER** ossia un server MQTT.

Tutti i dispositivi che intendono ottenere le temperature lette dal sensore, si iscrivono al broker ed è il medesimo che rende disponibile quanto richiesto quando viene effettuata una nuova lettura della temperatura.

La cosa veramente importante da sottolineare è che chi produce il dato non debba essere direttamente connesso con chi lo utilizza. Inoltre chi produce il dato non deve per forza di cose essere sincronizzato con il suo utilizzo.

La seguente figura illustra lo schema di massima appena descritto:



Topics

Un broker può gestire un numero molto elevato di messaggi, provenienti da diverse sorgenti e destinati a diversi utilizzatori. Come avviene il corretto smistamento dei messaggi? Tramite il concetto di **argomento** (*topic*in inglese): ogni messaggio viene pubblicato indicandone il topic (nel caso di un sensore di temperatura ad esempio il topic potrà essere appunto *temp*). In fase di sottoscrizione, l’utilizzatore deve specificare uno o più topic dei quali vuole ricevere i dati (sempre nel nostro esempio, per ricevere i dati di temperatura il termostato dovrà sottoscrivere il topic *temp*):

Per installare Mosquitto è sufficiente entrare nel seguente sito internet:

<https://mosquitto.org/>





Una volta avviato il server Mosquitto dovrebbe comparire tale finestra DOS priva di errori e dovrebbe rimanere così fino a quando non si stoppa tale server.



L’esempio seguente mostra come creare un semplice client in Python per pubblicare una variabile su topic:

import paho.mqtt.client as mqtt

def on\_message(client, userdata, message):

 print("message received " ,str(message.payload.decode("utf-8")))

 print("message topic=",message.topic)

 print("message qos=",message.qos)

 print("message retain flag=",message.retain)

if (\_\_name\_\_ == '\_\_main\_\_'):

 broker\_address="127.0.0.1"

 print("Crea nuova istanza del client")

 client = mqtt.Client("INS1")

 client.on\_message=on\_message

 client.connect(broker\_address)

 client.publish("test/var1","var. di test")

