



IoT con MQTT + Mosquitto + Python

07/07/2018 (Actualizado 09/07/2018)



Apoyar vía Patreon

Apoyar vía Paypal

Contactar



Ver vídeo

Como ya es costumbre en este tipo de vídeo aprovecho para adjuntar en este post el código que he ido poniendo en la pantalla. Aunque no es necesario para entender el artículo si que es recomendable ver el vídeo porque en directo se ven las cosas mucho mas claras:

Introducción a MQTT y Mosquito

MQTT(Message Queuing Telemetry Transport) es un protocolo de comunicación ligero M2M(Machine2Machine). Diseñado para ser tremendamente ligero(Poco ancho de banda, poco consumo y pocos recursos para su funcionamiento). Es ideal para aplicaciones IoT(Internet of Things) ya que lo puede implementar desde un servidor cloud hasta una tostadora o una nevera(Aunque no es SOLO para este tipo de aplicaciones).

Eclipse **Mosquitto** es una implementación Open Source de este protocolo ampliamente usada.

Básicamente, la arquitectura de una aplicación que utilice MQTT tiene: Un servidor MQTT que hace de broker de mensajes(Donde se conectan los clientes que publican o se suscriben a topics), los clientes que pueden tener un rol publicador, subscriptor o ambos.

[Contactar](#)

Un publicador es un componente que se dedica a publicar mensaies a un



```
sudo aptitude install mosquitto  
  
sudo aptitude install mosquitto-clients
```

Para el resto de plataformas es igual de fácil(Un instalador para Windows, etc.).

Arrancar Mosquitto

Para arrancar mosquito necesitamos un fichero de configuración. El siguiente es el fichero de configuración que hemos usado en el vídeo:

```
listener 1883 127.0.0.1  
allow_anonymous true
```

Esta configuración, básicamente, pone a la escucha al servidor de MQTT Mosquitto en el puerto 1883 y la IP localhost(127.0.0.1). También configuramos que los clientes se puedan conectar anonimamente.

Para arrancar mosquitto con esta configuración ejecutaremos:

```
mosquitto -c mosquitto.conf
```

Configurar un subscriptor

Mosquito nos provee de dos utilidades para testear todo. Una que nos permite subscribirnos a topics y otra que nos permite publicar en topics.

Si queremos configurar un subscriptor del tópic 'casa/cocina/nevera' lo podemos hacer de la siguiente manera:

```
mosquitto_sub -h 127.0.0.1 -p 1883 -t "casa/cocina/nevera" -v
```

Contactar



Aplicación Python para subscribirse a un topic

Tal como se ve en el vídeo para crear una aplicación Python utilizaremos la librería 'paho-mqtt' que instalaremos usando 'pip':

```
pip install --user paho-mqtt
```

Y esta es la aplicación:

```
import ssl
import sys

import paho.mqtt.client

def on_connect(client, userdata, flags, rc):
    print('connected (%s)' % client._client_id)
    client.subscribe(topic='casa/cocina/nevera', qos=2)

def on_message(client, userdata, message):
    print('-----')
    print('topic: %s' % message.topic)
    print('payload: %s' % message.payload)
    print('qos: %d' % message.qos)

def main():
    client = paho.mqtt.client.Client(client_id='albert-sub')
    client.on_connect = on_connect
    client.on_message = on_message
    client.connect(host='127.0.0.1', port=1883)
```

Contactar



Los comentarios están deshabilitados.

Podéis usar las redes sociales o el formulario de contacto para comentar.

Suscríbete y recibe formación, consultoría y desarrollo gratis en tu mail

[Política de privacidad](#)



LOSTSYS

Mecenas
Gestión IT

- Anónimo1

Contactar

- Tel. 610 459 269
- Formulario de contacto

Formación

- Formación Liferay
- Formación Docker y Kubernetes
- Formación Microservicios
- Formación Elasticsearch
- Formación Apache Kafka



© 2005 - 2022 Albert Coronado Calzada - Política de privacidad

Contactar