

Seminario 5: Protocolo MQTT

Sistemas Distribuidos

Salvador Gutiérrez Salcedo

Departamento de Ingeniería Informática
Universidad de Cádiz



Curso 2019 – 2020

Indice

- 1 Sistemas publicador-suscriptor
- 2 El protocolo MQTT
- 3 Mosquitto e implementación en Python
- 4 Ejemplo práctico

Sección 1 | Sistemas publicador-suscriptor

Sistemas publicador-suscriptor

- Sistema donde los **publicadores** publican eventos estructurados a un servicio de eventos
- Y los **suscriptores** expresan interés por ciertos eventos, a través de **suscripciones** que pueden ser patrones sobre estos eventos estructurados
- Ejemplo: un suscriptor podría estar interesado en todos los eventos relacionados con noticias de la ciudad de Cádiz
- El objetivo principal de estos sistemas es relacionar las suscripciones con los eventos publicados y asegurar la entrega correcta de las **notificaciones de eventos**

Aplicaciones de sistemas publicador-suscriptor

Gran variedad de dominios de aplicación, especialmente relacionados con diseminación de eventos a gran escala

- Sistemas de información financieros
- Flujo de datos en tiempo real: RSS *feeds*, **plataformas IoT**, ...
- Apoyo al trabajo colaborativo: los participantes necesitan estar informados de eventos de interés compartido
- Apoyo a la computación ubicua, incluyendo la gestión de eventos que provienen de infraestructuras ubicuas (eventos de localización...)
- Aplicaciones de monitorización, incluyendo monitorización de redes en Internet
- Componente clave de la infraestructura de Google (diseminación de eventos sobre anuncios, *ad clicks*, a las partes interesadas)

Aspectos de implementación



Un modelo de programación centralizada **podría implementarse mediante la creación de un *broker* o agente**. Los publicadores envían eventos a este agente, y los suscriptores envían suscripciones también al *broker* para recibir notificaciones

Aspectos de implementación

- La interacción con el agente es mediante una serie de mensajes punto a punto, implementados mediante paso de mensajes o invocación remota
- Poco escalable: el agente puede fallar y además se convierte en un cuello de botella

Sección 2 | El protocolo MQTT

MQTT

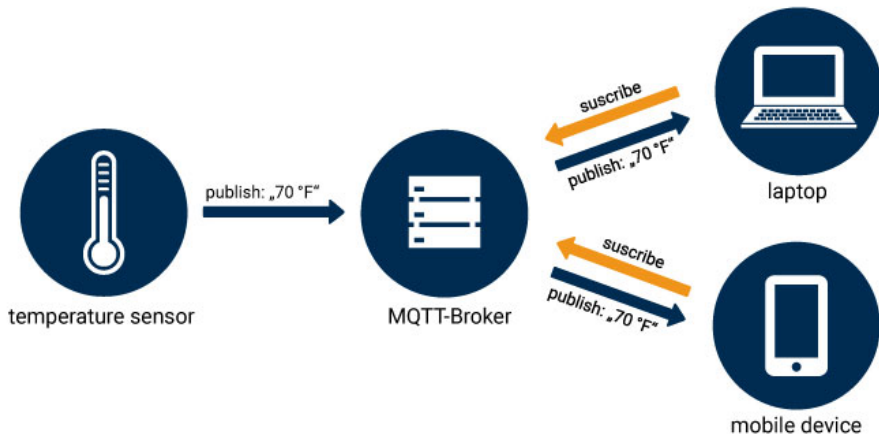


- Es un protocolo publicador-suscriptor para el envío de mensajes entre dispositivos
- Sus siglas vienen de *Message Queing Telemetry Transport*
- Suele operar sobre el protocolo TCP/IP
- Se utiliza ampliamente en aplicaciones IoT

Broker MQTT

- Es el elemento central del protocolo publicador-suscriptor implementado por MQTT
- Se encarga de recibir todos los mensajes, filtrarlos, comprobar las suscripciones, y enviar los mensajes a cada suscriptor concreto
- Otra responsabilidad del broker es la autenticación y autorización de los clientes

MQTT en la *Internet of Things*



Sección 3 | Mosquitto e implementación en Python

Mosquitto



- Es una implementación de un broker MQTT
- Desarrollado por la *Eclipse Foundation*, Mosquitto se diseñó como un elemento ligero, y adecuado para su uso en dispositivos desde bajo consumo y potencia hasta servidores completos
- Se puede descargar desde <https://mosquitto.org/download/> o desde los repositorios de muchas distribuciones

paho



- Es una implementación de un cliente MQTT, también desarrollada por la *Eclipse Foundation*
- Es una librería pensada especialmente en la *Internet of Things* (dispositivos ligeros, con poco consumo, poca capacidad de cómputo...)

Esquema



Sección 4 | Ejemplo práctico