

Envío y recepción de mensajes MQTT con pub/sub en Node -Red

Elían Toapanta, Iván Zambrano y Marlon Torres

Resumen – En este documento va a bordar el protocolo de comunicación MQTT (Message Queue Telemetry Transport) en Node-Red.

Keywords – MQTT, Node-Red, nodos.

I. INTRODUCCION

MQTT (Message Queue Telemetry Transport), es un protocolo usado para la comunicación machine-to-machine (M2M) en el "Internet of Things". Este protocolo está orientado a la comunicación de sensores, debido a que consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos (CPU, RAM, ...). Un ejemplo de uso de este protocolo es la aplicación de Facebook Messenger tanto para android y Iphone.

La IEEE hará el último formato de su documento. Si el documento está destinado a una conferencia, por favor tenga en cuenta el numero limite de paginas.

II. NODE-RED

A. Definición

Node-RED es una herramienta de programación visual. Muestra visualmente las relaciones y funciones, y permite al usuario programar sin tener que escribir una lengua. Node-RED es un editor de flujo basado en el navegador donde se puede añadir o eliminar nodos y conectarlos entre sí con el fin de hacer que se comuniquen entre ellos.

En Node-RED, cada nodo es uno de los siguientes dos tipos: un nodo de inyección o un nodo de función. Los nodos de inyección producen un mensaje sin necesidad de entrada y lanzan el mensaje al siguiente nodo conectado a éste. Los nodos de función, por el contrario, tienen una entrada y realizan algún trabajo en él. Con una gran cantidad de estos nodos para elegir, Node-Red hace que el conectar los dispositivos de hardware, APIs y servicios en línea sea más fácil que nunca.

B. Nodos utilizados

Mqtt in:

Recibe los datos del estado de la puerta que le envía el publicador. Su configuración es la de colocar la dirección IP, su puerta y en la parte del security colocamos las credenciales del servidor.

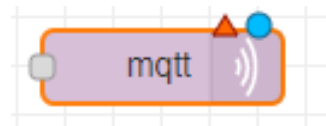


Fig. 1. Nodo "Mqtt in" en NODE-RED.

Mqtt out:

Hace la propiedad del publicador el publica el estado del foco. Su configuración es la de colocar la dirección IP, su puerta y en la parte del security colocamos las credenciales del servidor.

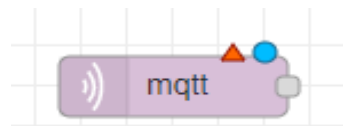


Fig. 2. Nodo "Mqtt out" en NODE-RED.

Audio out:

Muestra los datos del publicador que es la puerta mediante un sonido, por medio del dashboard. Su configuración es de conectar con el switch el cual tiene el nombre de “Puerta”, luego elegimos el Grupo que es “[Proyecto_investigacion] MQTT”.

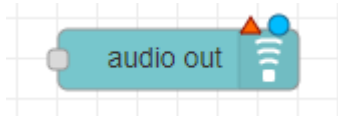


Fig. 3. Nodo “audio out” en NODE-RED.

Text:

Es solo para colocar un título el cual tiene el nombre de “Alarma” Su configuración es de conectar con el switch el cual tiene el nombre de “Puerta”, luego elegimos el Grupo que es “[Proyecto_investigacion] MQTT”.

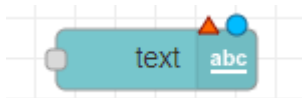


Fig. 4. Nodo “text” en NODE-RED.

Show notification:

Muestra en el dashboard el estado de la puerta mediante una notificación la cual aparece en la parte superior de la pantalla Su configuración es de conectar con el switch el cual tiene el nombre de “Puerta”, luego elegimos el Grupo que es “[Proyecto_investigacion] MQTT” y cambiamos el Label a Alarma.

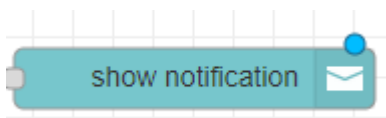


Fig. 5. Nodo “show notification” en NODE-RED.

Switch:

Sirve para representar dos estados, uno que es el on Payload u off Payload, el cual esta modificado para que envíe un texto diciendo “abierta” o “cerrada”, luego elegimos el Grupo que es “[Proyecto_investigacion] MQTT”.

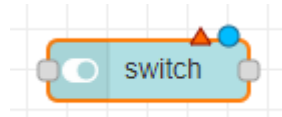


Fig. 6 Nodo “switch” en NODE-RED.

III. MQTT

MQTT son las siglas MQ Telemetry Transport, aunque en primer lugar fue conocido como Message Queing Telemetry Transport. Es un protocolo de comunicación M2M (machine-to-machine) de tipo message queue.

Está basado en la pila TCP/IP como base para la comunicación. En el caso de MQTT cada conexión se mantiene abierta y se "reutiliza" en cada comunicación. Es una diferencia, por ejemplo, a una petición HTTP 1.0 donde cada transmisión se realiza a través de conexión.

MQTT fue creado por el Dr. Andy Stanford-Clark de IBM y Arlen Nipper de Arcom (ahora Eurotech) en 1999 como un mecanismo para conectar dispositivos empleados en la industria petrolera.

A. Funcionamiento

El funcionamiento del MQTT es un servicio de mensajería push con patrón publicador/suscriptor (pub-sub). Como vimos en la entrada anterior, en este tipo de infraestructuras los clientes se conectan con un servidor central denominado broker.

Para filtrar los mensajes que son enviados a cada cliente los mensajes se disponen en topics organizados jerárquicamente. Un cliente puede publicar un mensaje en un determinado topic. Otros clientes pueden suscribirse a este topic, y el broker le hará llegar los mensajes suscritos.

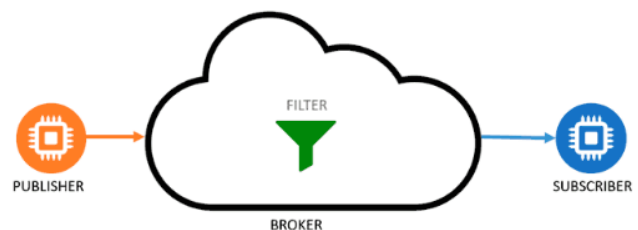


Fig. 7 Diagrama simple del funcionamiento de MQTT.

Los clientes inician una conexión TCP/IP con el broker, el cual mantiene un registro de los clientes conectados. Esta conexión se mantiene abierta hasta que el cliente la finaliza. Por defecto, MQTT emplea el puerto 1883 y el 8883 cuando funciona sobre TLS.

B. Los clientes MQTT

Los clientes MQTT pueden abarcar un amplio rango de formatos. Pueden ser clientes MQTT que recolectan información del medio (sensores y sistemas embebidos) o aplicaciones ejecutando alguna librería MQTT y que de alguna forma interactúen con los datos. Pueden ser divulgadores (publicadores) y subscriptores de mensajes y además, pueden controlar y configurar los sensores a su cargo mediante comandos, si es que son nodos de sensores.

Estos siempre se conectan a un tercer participante, denominado broker de mensajes.

C. El bróker MQTT

El broker MQTT es un servicio (software) que implementa el protocolo MQTT y que establece la comunicación, a nivel de aplicación, entre los diferentes clientes. Hace de intermediario entre los productores y los consumidores. Es el responsable de recibir los mensajes, filtrarlos y rutarlos a los clientes suscritos según su topic. Otra tarea importante del brokers es autorizar el acceso e identificar los clientes. Puede haber varios brokers en una misma red.

D. PUB/SUB

Pub/Sub es un servicio de mensajería asíncrono que separa los servicios que producen eventos de los que procesan eventos. Puedes usar Pub/Sub como la transferencia y entrega de eventos o de software multimedia orientado a la mensajería para las canalizaciones de estadísticas de transmisión.

Pub/Sub ofrece almacenamiento duradero de mensajes y entrega de mensajes en tiempo real con alta disponibilidad y rendimiento uniforme a gran escala.

E. Seguridad en MQTT.

La seguridad siempre debe ser un factor importante a considerar en cualquier sistema de comunicación M2M. El protocolo MQTT dispone de distintas medidas de seguridad que podemos adoptar para proteger las comunicaciones.

Esto incluye transporte SSL/TLS y autenticación por usuario y contraseña o mediante certificado. Sin embargo, hay que tener en cuenta que muchos de los dispositivos IoT disponen de escasa capacidad, por lo que el SLL/TLS puede suponer una carga de proceso importante.

En muchos casos, la autenticación consiste en una contraseña y usuario que son enviados como texto plano. Por último, también es posible configurar el broker para aceptar conexiones anónimas.

Todo esto debe ser tenido en cuenta a la hora de configurar un sistema MQTT, y entender los riesgos de cada uno de ellos, así como su impacto en la eficiencia del sistema.

F. Ventajas de MQTT.

Otra consecuencia de la ligereza del protocolo MQTT es que requiere un ancho de banda mínimo, lo cual es importante en redes inalámbricas, o conexiones con posibles problemas de calidad.

Por último, MQTT dispone de medidas adicionales importantes, como la seguridad y calidad del servicio (QoS). Por último, es una solución largamente testada y consolidada, que aporta robustez y fiabilidad.

IV. ESP8266

El ESP8266 se puede integrar en un WiFi que provee un chip de bajo coste con una pila TCP/IP completa y un microcontrolador. Se alimenta con 3.3v y dispone de un procesador Tensilica Xtensa LX106 de 80 Mhz, memoria RAM de 64 KB para instrucciones y 96 KB para datos, 16 pines GPIO, pines dedicados UART, e interfaz SPI y I2C.

En cuanto a las capacidades de conectividad WiFi, es compatible con el estándar IEEE 802.11 b/g/n, además de soportar seguridad WEP, WPA y WPA2.

El ESP8266, dicho de un modo simple, agrega capacidad de conectividad WiFi a nuestros proyectos. Es decir, permite conexión wireless a una red local o a Internet. Eso posibilita gran cantidad de posibilidades, como poder conectar o desconectar electrodomésticos u otro

tipo de sistemas mecánicos de nuestro hogar para domotizar la casa y controlarla por Internet desde nuestro smartphone o cualquier ordenador conectado desde cualquier parte.



Fig. 8 ESP8266.

V. CONCLUSIÓN

Otra consecuencia de la ligereza del protocolo MQTT es que requiere un ancho de banda mínimo, lo cual es importante en redes inalámbricas, o conexiones con posibles problemas de calidad.

Por último, MQTT dispone de medidas adicionales importantes, como la seguridad y calidad del servicio (QoS). Por último, es una solución largamente testada y consolidada, que aporta robustez y fiabilidad.

El protocolo MQTT se ha alzado como uno de los estándares para aplicaciones IoT tanto comerciales como de ámbito maker. Por supuesto, hay muchos más aspectos de los que podríamos hablar mucho más sobre MQTT, como funciones avanzadas de seguridad, permanencia de los mensajes en el broker, configuración de varios brokers.

REFERENCES

- [1] Granado Rodríguez, I. (2018). Integración en la plataforma SmartSantander de un sensor PM2.5 basado en MQTT. Semle, A., & eFalcom, K. (2016). Protocolos IIoT para considerar. *Revista AADECA*.
- [2] De Giusti, A. E., Rodriguez, I. P., Costanzo, M., & Boggia, M. (2017). Cloud robotics: Auto Rover 4WD y cuadricóptero controlados remotamente desde AWS. In *XXIII Congreso Argentino de Ciencias de la Computación (La Plata, 2017)*.
- [3] Nuñez-Agurto, D., Benavides-Astudillo, E., Rodríguez, G., & Salazar, D. (2020). Propuesta de una plataforma de bajo costo basada en Internet de las Cosas para Agricultura Inteligente. *Cumbres*, 6(1), 53-66.
- [4] Woo, F. D. J. B., García, A. G., del Real, T. A. R., & Sánchez, E. J. A. (2018). SISTEMA DE ADQUISICIÓN DE DATOS DE BAJO COSTO PARA UN INVERNADERO BASADO EN TECNOLOGÍA DE ACCESO LIBRE. *Pistas Educativas*, 39(128).
- [5] Palaguachi Encalada, S. I. (2018). *Diseño, desarrollo e implementación de una estación meteorológica basada en una red jerárquica de sensores, software libre y sistemas embebidos para la Empresa ELECAUSTRO en la Minicentral Gualaceo utilizando comunicación MQTT y MODBUS* (Bachelor's thesis).
- [6] Villena, M., Serrano, V., Hoyos, D., & Zutara, F. IMPLEMENTACIÓN DE UNA RED DE SENSORES INALÁMBRICA MEDIANTE SISTEMAS EMBEBIDOS Y MQTT PARA EL REGISTRO DE VARIABLES FÍSICAS DE INTERÉS EN ENERGÍAS RENOVABLES.
- [7] Campoverde, A. M., Hernández, D. L., & Mazón, B. E. (2015). Cloud computing con herramientas open-source para Internet de las cosas. *Maskana*, 6, 173-182.
- [8] López, R. V. (2020). Propuesta de una estación móvil para la medición de la calidad del aire con el Internet de las Cosas. *Revista Multidisciplinaria de Avances de Investigación*, 6(1), 1-12.