

EA076 – LABORATÓRIO DE SISTEMAS EMBARCADOS

Rede de IoT baseada em MQTT

Prof. Antonio Quevedo e Profa. Wu Shin Ting

MQTT, acrônimo de *Message Queuing Telemetry Transport*, é um protocolo de mensagens leves para redes de pilha de protocolos TCP/IP [1]. Hoje em dia, ele é o mais utilizado para implementações IoT (*Internet of Things*). Este protocolo define dois tipos de objetos na rede: 1 servidor *broker* e n clientes. Toda a interação é feita através de mensagens de inscrição (*subscribe*)/publicação (*publish*) via o servidor. Ao receber uma mensagem inscrita em um tópico de um cliente, o servidor roteia esta mensagem para todos os clientes que estão inscritos neste tópico específico.

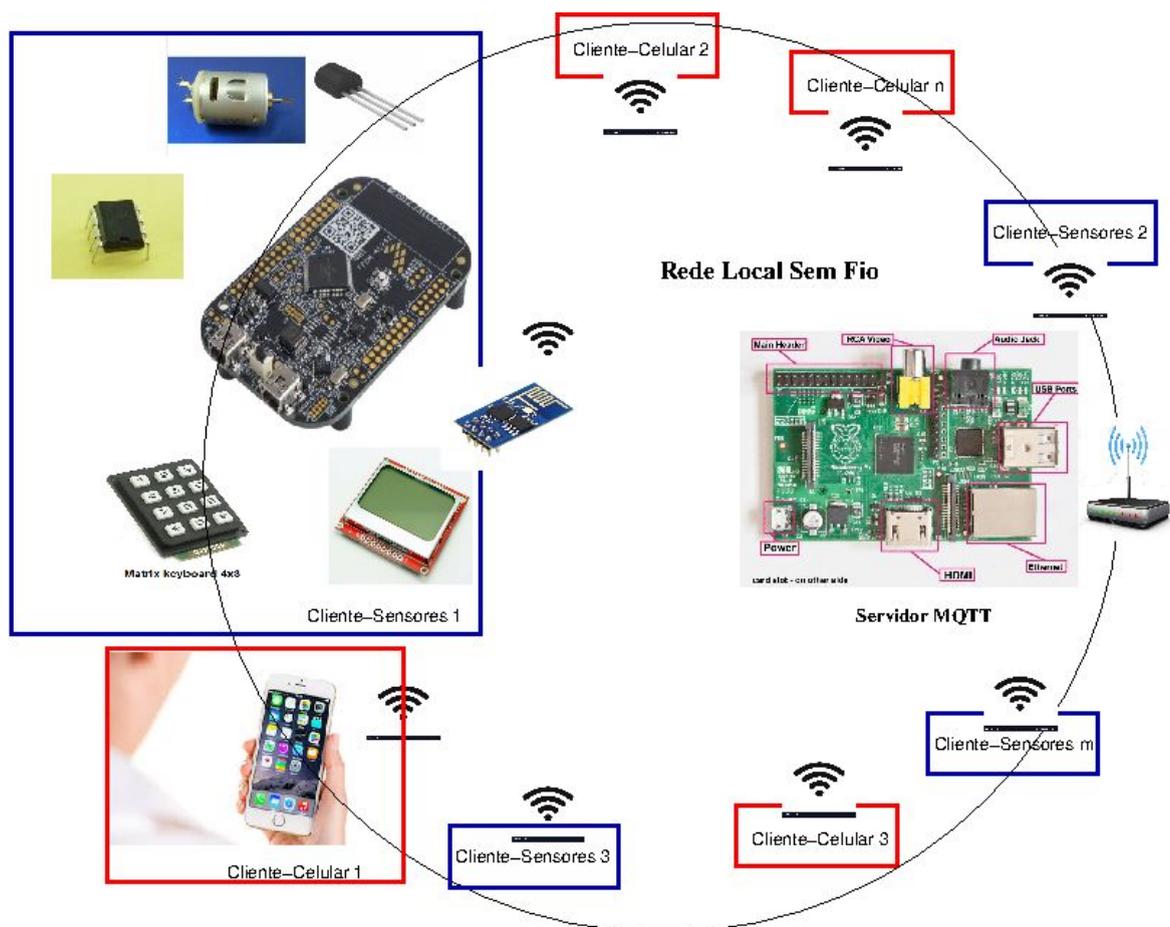


Figura 1: Rede WiFi EA076 com protocolo MQTT para o curso EA076 do 1o. Semestre de 2019.

Uma implementação popular da versão 3.1 do MQTT é o Mosquitto [2]. Para construir uma infra-estrutura de rede (local) local (LAN) WiFi baseada em MQTT no nosso curso, (1) configuramos um roteador sem fio como ponto de acesso de uma rede local sem fio, cujo SSID (*Service Set Identifier*) é EA076 e senha FRDMKL25, (2) conectamos um Raspberry Pi a este roteador, (3) configuramos o Raspberry com um IP estático 192.168.1.111 [3] e (4) instalamos nele

o servidor Mosquitto MQTT [4]. Foi mantido o número *default* da porta do Mosquitto que é 1883. Foi ainda configurada uma senha de acesso ao *broker*: *username*: aluno e senha: UNICAMP [5]. Com isso, qualquer dispositivo conectado à rede sem fio e que tem um programa de cliente MQTT instalado, como os clientes-sensores e clientes-celulares mostrados na Figura 1, pode se comunicar com o servidor MQTT (*broker*) através do seu IP estático. O conjunto Raspberry e roteador se encontra na sala de SATE. E neste curso, vamos aprender como implementar um cliente MQTT com uso de nosso microcontrolador e um módulo WiFi ESP-01.

Para que o nosso microcontrolador se torne um cliente MQTT, utilizamos os módulos WiFi ESP-01 [6] como interface entre o microcontrolador e a rede WiFi, e gravamos neles um *firmware* do cliente MQTT [7]. Este *firmware* é uma versão simplificada dos códigos disponíveis em [8]. Os comandos implementados estão sintetizados em [5].

Para que os celulares ou *smartphones* se comuniquem com o broker, basta instalar neles um dos dezenas de aplicativos de cliente MQTT disponíveis.

Referências

[1] IBM. *MQTT V3.1 Protocol Specification*.

ftp://ftp.dca.fee.unicamp.br/pub/docs/ea076/manuais/MQTT_V3.1_Protocol_Specific.pdf

[2] Eclipse Mosquitto. An open source MQTT broker. <https://mosquitto.org/>

[3] Salim Aouar. Configurando IP estático no Raspberry Pi com o sistema Raspbian.

<https://www.salimaouar.com.br/artigo/configurando-ip-estatico-no-raspberry-pi-com-o-sistema-raspbian>

[4] Oneguyoneblog. *IOT Server: Mosquitto and Node Red on Raspberry Pi*.

<https://oneguyoneblog.com/2017/06/20/mosquitto-mqtt-node-red-raspberry-pi/>

[5] A. Quevedo. ESP8266 Cliente MQTT.

ftp://ftp.dca.fee.unicamp.br/pub/docs/ea076/complementos/ESP8266_CLIENTE_MQTT.pdf

[6] Microchip. ESP-01 WiFi Module. <http://www.microchip.ua/wireless/esp01.pdf>

[7] A. Quevedo. MQTT_Client.ino.

ftp://ftp.dca.fee.unicamp.br/pub/docs/ea076/complementos/MQTT_Client.ino

[8] nopnop2002. MQTT via ESP01. https://github.com/nopnop2002/MQTT_via_ESP

[9] MyMQTT. https://play.google.com/store/apps/details?id=at.tripwire.mqtt.client&hl=pt_BR

[10] IoT MQTT Panel.

<https://play.google.com/store/apps/details?id=snr.lab.iotmqttpanel.prod&hl=pt>