

Protocolo MQTT para sistemas de IoT

Um estudo técnico/prático

Larissa L. Wong Marco A. G. Pedroso Victor E.
Almeida

UNIOESTE

21 de abril de 2022



Conteúdo

- 1 Introdução
- 2 História
- 3 Prática
- 4 Conclusão



Colocar mais algo para introduzir

TODO



Criação do Protocolo

Começou a ser projetado durante a década de 1990 por:



Figura 1: Andy Stanford-Clark da IBM



Figura 2: Arlen Nipper da Cirrus Link/Eurotech



Problemas a serem resolvidos

- Resolver o problema de conexão de oleodutos via satélite.
- Limitações:
 - Alta latência;
 - Baixa largura de banda;
 - Dispositivos com pouca bateria.



Requisitos do Protocolo

- Implementação simples;
- Uso de QoS, *Quality of Service* por quem publica a mensagem;
- Uso eficiente de largura de banda, baixo *overhead*;
- Baixo custo energético para envio;
- Possibilidade de enviar qualquer tipo de dado;
- Possibilidade de manter conexões ativas, prontas para enviar e receber dados;



Fase do protocolo proprietário



- Primeira versão implementada no ano de 1999;
- Batizado MQTT, *MQ Telemetry Transport*, em referência ao produto da IBM MQ Series
- Muito utilizado embarcado em produtos da IBM.



Fase do protocolo aberto I

- Demanda/Aplicabilidade IoT;
- Em 2010 o protocolo se tornou livre;
- Primeira versão lançada 3.1;
- Investimentos da IBM através da Eclipse Foundation para criar um ecossistema em torno do protocolo.



Fase do protocolo aberto II



Figura 3: Exemplos de aplicações do ecossistema MQTT



Fase do protocolo aberto III

- No ano de 2013 a IBM buscou padronização com a OASIS;
- 29 de outubro de 2014 o MQTT foi aprovado como padrão pela OASIS na sua versão 3.1.1



Fase atual do protocolo

- A última versão 5.0 março de 2019;
- Funcionalidades modernas como:
 - facilidade de conexão e interação com a nuvem;
 - Tratamento de erros;
- Implementações de clientes para diversos sistemas e linguagens;
- Implementação de diversos brokers;
- Utilizado por grandes empresas tanto software aberto quanto proprietário.



Dispositivos e softwares da parte prática

- Esp32:
 - Sensor Adafruit BMP-280;
 - Relay para ligar lâmpada;
 - C/C++ (Framework Arduino e ESP-IDF);
- Raspberry:
 - Docker executando o broker mosquitto;
 - Cliente inscrito no tópico “#”



Diagrama da aplicação

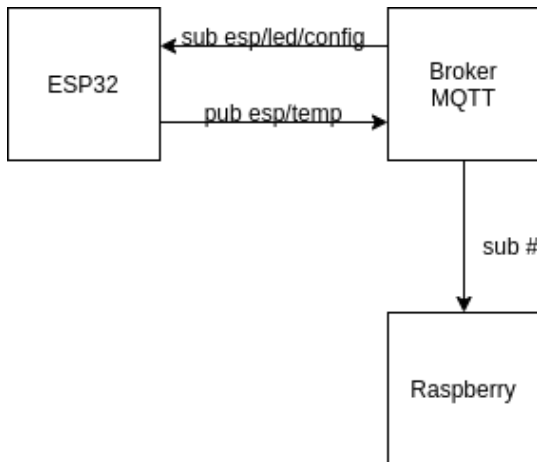


Figura 4: Dispositivos e tópicos utilizados



Mão na massa!!



Agradecimentos

Perguntas?



Obrigado pela atenção

