

Sistemas de Comunicação Móvel, 3º Trabalho de Laboratório:

Os ficheiros correspondentes ao código das primeira e segunda metas encontram-se disponíveis nas pastas Meta 1 e Meta 2 respetivamente.

Descrição do sistema desenvolvido:

Arquitetura Geral e Descrição do Sistema:

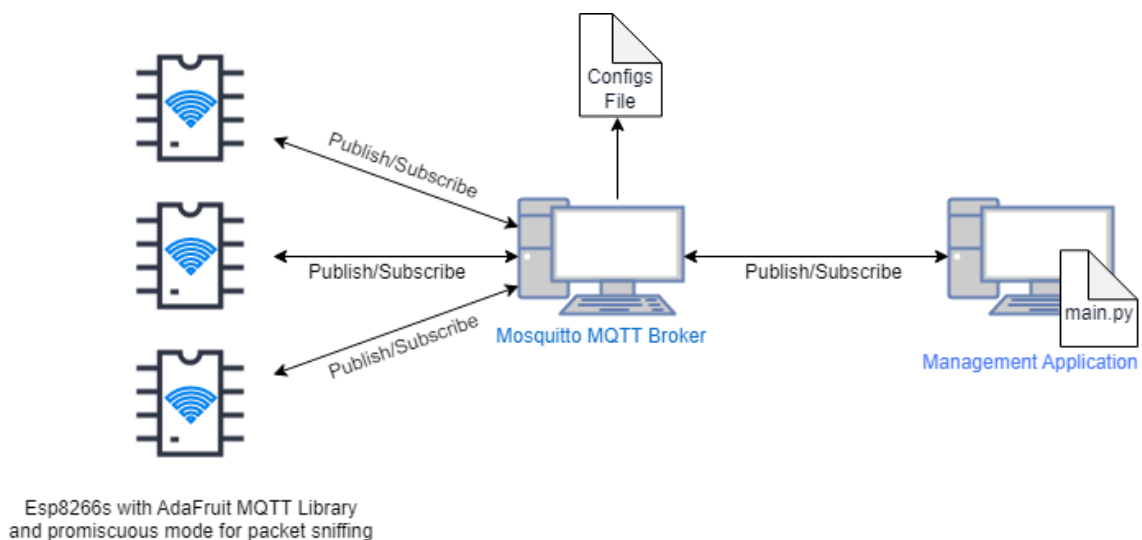


Figura 1 – Arquitetura geral do sistema desenvolvido

Como é possível observar na **Figura 1**, o sistema é constituído por vários Arduinos, ambos com as bibliotecas ESP8266, com capacidades de WiFi e AdaFruit MQTT, para permitir a conexão dos dispositivos a um MQTT broker. Também tomam partido do modo promiscuo para conseguir detetar os pacotes transmitidos numa rede (código baseado no repositório seguinte: <https://github.com/n0w/esp8266-simple-sniffer/tree/master/src>). Todos estes dispositivos conectam-se por WiFi a um Access Point definido por um utilizador e por MQTT a um broker (no âmbito deste projeto recorreremos ao Mosquitto MQTT), que servirá de ponte entre estes Arduinos e uma aplicação desenvolvida em Python, com o intuito de fornecer uma lista de endereços a pesquisar e imprimir na consola os pacotes desses endereços encontrados.

Comportamento esperado:

Primeiramente deve-se ligar o Mosquitto Broker, executando o mesmo através do seguinte comando (o ficheiro `test.conf` deve estar inserido na diretoria do mosquitto):

```
mosquitto -v -c test.conf
```

De seguida, ligar o(s) Arduino(s) e fazer as alterações necessárias para correr o código na rede do utilizador, com o nome que pretender dar a cada dispositivo. Ao ligar deve de seguida conectar a aplicação Python e fornecer o(s) endereço(s) a pesquisar. Os Arduinos recebem a informação e comunicam à aplicação quando um pacote desejado é encontrado, ligando o seu LED durante dois segundos e enviando um pacote de informação contendo o RSSI, MAC Address, canal encontrado e o tópico do dispositivo que encontrou.

Para assegurar a conexão de múltiplos Arduinos, sempre que um novo se conecta, envia um publish à aplicação a pedir a lista de endereços. A aplicação pode então enviar uma nova lista, o que pode ser bastante útil quando se pretende procurar em Arduinos diferentes pacotes diferentes. O número máximo de endereços é 20.

Descrição das soluções MQTT:

Para o correto funcionamento do sistema, temos vários tópicos criados para garantir a transmissão dos dados.

Arduino:

- **SUBSCRIBE** ao tópico `scm/pl3/addresses` para receber a lista de endereços por parte da aplicação.
- **PUBLISH** aos tópicos `scm/pl3/newDevices` para enviar uma mensagem à aplicação que acabou de se conectar e `scm/pl3/sniffer/{nome}` para enviar a informação do pacote encontrado (o parâmetro nome deve ser definido no início do código do Arduino).

Aplicação:

- **SUBSCRIBE** aos tópicos `scm/pl3/newDevices` e `scm/pl3/sniffer/#`, para assegurar uma subscrição MQTT wild-card e assim poder receber os pacotes de informação de todos os dispositivos.
- **PUBLISH** ao tópico `scm/pl3/addresses`.

Evitar o rastreamento em plataformas móveis:

A partir do iOS14 a Apple implementou um sistema de segurança que se baseia em utilizar um endereço MAC diferente para diferentes redes WI-Fi. Com o iOS15 vieram também outros métodos de proteção do MAC, tais como: se um dispositivo não se conectar durante 6 semanas, o endereço privado será diferente na próxima conexão; é também possível definir o dispositivo para trocar de endereço numa rede de 2 em 2 semanas.

Referencias

- <https://support.apple.com/pt-pt/HT211227>