



TkkrLab

# MQTT on ESP32

IoT in de praktijk / Kamiel Straatman

(Zaterdag 19 November 2023, 11 slides)

Disclaimer: Deze sessie is gesponsord door HiveMQ GmbH die elke deelnemer een ESP-32 device ter beschikking stelt en voorziet in gratis MQTT cloud services. De inhoud van de training is echter objectief en primair gericht op het leren programmeren van een ESP-32 in C en het gebruik van de open standaard 'MQTT'.



# Communicatie is vaak 1 op 1.

Mens tot mens : Spraak, tekst, (zang), brief\*

Mens tot computer : Webbrowser, (GUI) / HTTP

'Computer' tot 'Computer' (ofwel IoT 'devices')

: API (advanced programming interface)

Maar werkt een API wel goed ?

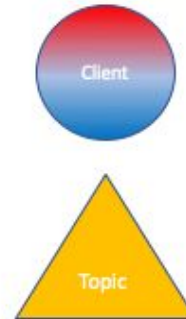
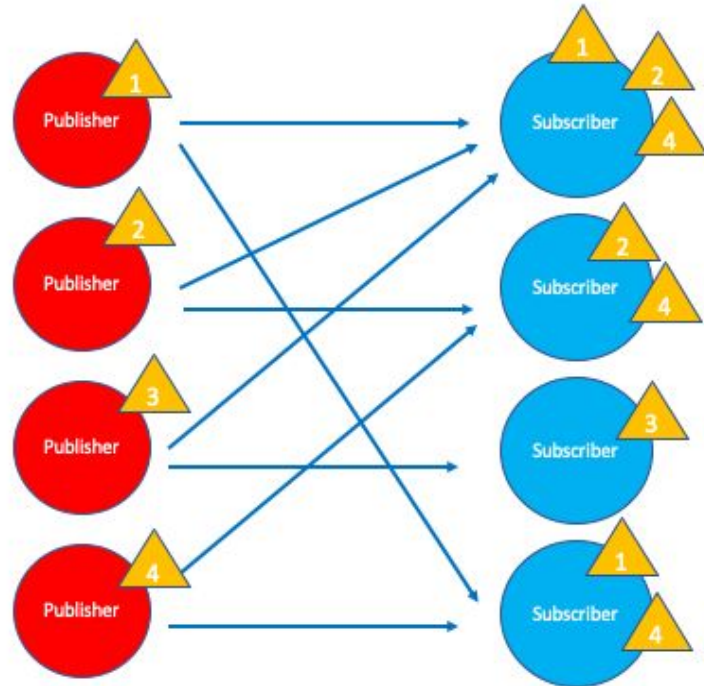
: Nee, niet altijd. PUB/SUB\* en dus MQTT is soms beter.

Wanneer MQTT : Slechte netwerken, Simpele of snelle computers: 'things' etc.



# Een op een relaties (APIs)

Veel verbindingen allemaal door elkaar....

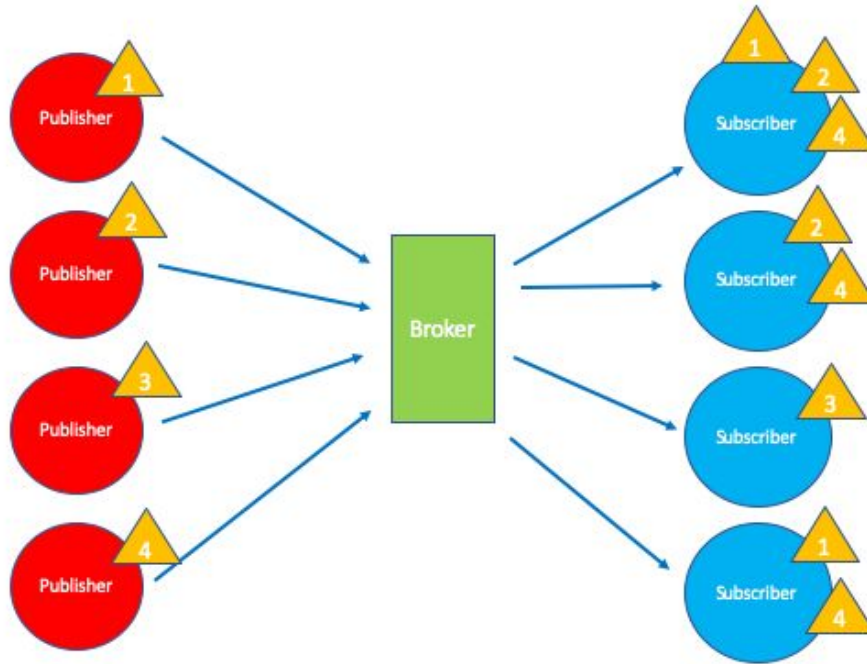


“Zenders” moeten hun ontvangers “kennen”; Ze moeten precies hun onderlinge ‘Taal’ spreken en dat ‘gelijkgestemd’.

- En wat te doen als een partner even ‘stil’ is of wegvalt ?
- API zijn soms maar lang niet altijd de oplossing.

# En nu de 'man in het midden': de MQTT 'broker' :

Een postkantoor in het midden ..



Een zender hoeft alleen maar de volgende zaken te weten:

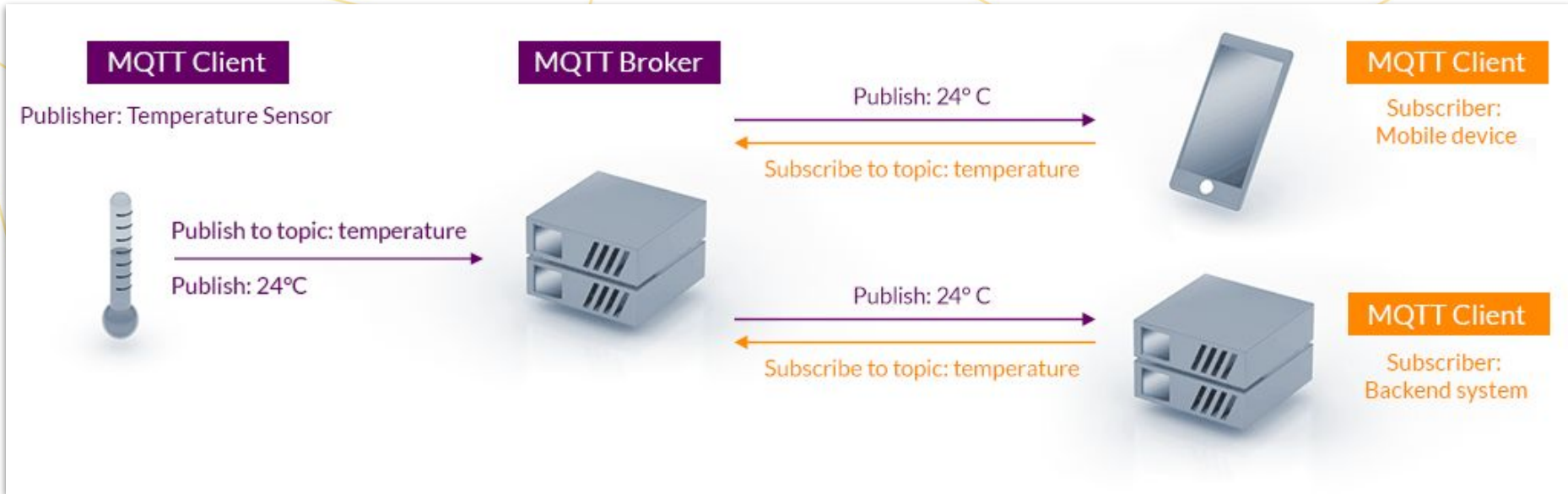
- Het adres van het postkantoor oftewel de **broker** (IP)
- Het onderwerp of **topic** waarnaar te '**publiceren**'

De ontvanger zender hoeft alleen maar de volgende zaken te weten:

- Het adres van het postkantoor oftewel de **broker** (IP)
- Het onderwerp of **topic** waarnaar te abonneren oftewel te '**Subscriben**'

# PUB / SUB architectuur :

MQTT is Pub / Sub gebaseerd



# MQTT op je laptop

GUI	:	EMQX of MQTT explorer
CLI	:	<b>HiveMQ CLI</b> of Mosquitto CLI
Topics	:	topic/subtopic1/.../subtopicX (no leading slash)
Wildcards	:	1 level        "+" Multilevel    "#"

```
mqtt pub -h broker.hivemq.com -t  
"tkkrlab/kamiel" -m "hallo world"
```

Probeer ook es "-d" (=debug info)

<https://github.com/hivemq/mqtt-cli/releases/download/v4.22.0/mqtt-cli-4.22.0-win.zip>

```
mqtt sub -h broker.hivemq.com -t  
"tkkrlab/#"
```

Probeer ook es "-J"

of wildcards :    -t "tkkrlab/naam/#"  
                  -t tkkrlab/+/sensor"



# Wat gaan we doen ?

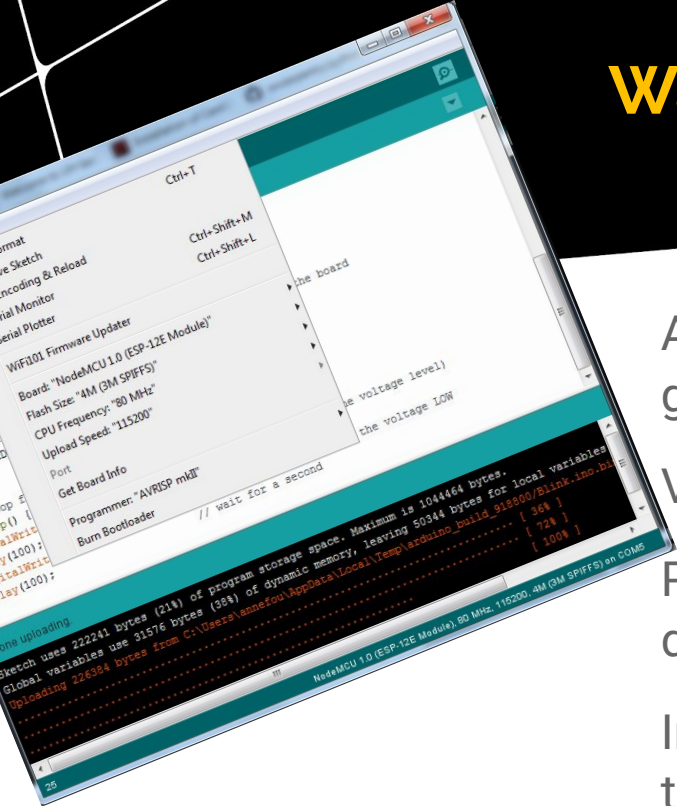
- Een device (T-display ESP-32 S3) programmeren zodat ie MQTT boodschappen kan verzenden en ontvangen.

Daarvoor is nodig:

- Het opzetten van de programmeeromgeving (Platform.IO) op je laptop.
  - Het bestuderen van de code
  - Het programmeren/uploaden naar je ESP-32
  - Het instellen van parameters (SSID, broker URL, topic etc) van je ESP-32
  - Het testen (in paren) van de functies.
- 
- *En dan aanpassen van de code (als dat nodig is) waarna de upload en test cyclus zich herhaald (DevOps)*

# Waarom Platform.IO ?

(en niet de Arduino IDE)



Arduino IDE is erg laagdrempelig maar net niet solide genoeg voor professioneel development.

Veel variatie in individuele installaties (per PC)

Platform.IO is stricter en vollediger in de software definities en laat dus minder ruimte voor toeval.

Integratie met VScode, en daarmee met vele andere tools, is superieur

FF iets lastiger maar uiteindelijk veel prettiger.





# Programma structuur

---

- Platform.ini
- Tkkr-MQTT-v2
  - FS Init (geduld)
  - Main.ino - Hoofdprogramma:
    - Setup
      - Housekeeping (Serial / PIN's etc)
      - Get / setup both Wifi / MQTT
      - Enable MQTT
      - *Hook: MQTT\_messageReceived*
    - Loop
      - MQTT.loop
      - Button(..)
  - Includes:
    - Pins.h
    - MQTT\_strings.h
    - Logo.h
    - Display-cmd.c

## Testing..

- Download VScode
- Installeer de Platform.io extension.
- Download / unzip de GIT repo:
  - <https://github.com/floresboy/T-Display-S3-MQTT>
- Open examples: `main.ino`
- Klik 'compile' or 'program'



- Connect met de wifi, check je 'topic' en 'broker' en connect een 'SUBscribe' sessie.
- Test en pas aan zoveel je wilt.

Dank jullie wel !

