Communication Protocols for IOT

Tiemon Steeghs

Inhoud

[Aanleiding 3](#_Toc131422363)

[Probleemstelling 3](#_Toc131422364)

[Hoofdvraag 4](#_Toc131422365)

[Deelvraag 1 4](#_Toc131422366)

[Deelvraag 2 4](#_Toc131422367)

[Deelvraag 3 4](#_Toc131422368)

# Aanleiding

Om beter te begrijpen welke IOT communicatie protocollen er zijn en hoe ze werken ga ik dit onderzoek voeren. Hierbij ligt de focus op de CoAp en de MQTT protocollen.

# Probleemstelling

Ik weet op dit moment nog niet precies welke IOT communicatie protocollen er zijn en wat hun toepassingen zijn. Dit ga ik met behulp van dit onderzoek oplossen.

# Hoofdvraag: Welk IOT protocol kan je het best gebruiken voor smart-home temperatuur regeling en hoe zou dit in werking gaan?

## Deelvraag 1 Wat is IOT?

IOT of internet of things gaat over de interconnectie van verschillende embedded systemen waardoor deze systemen gegevens kunnen verzamelen en uitwisselen. De gegevens kunnen vervolgens worden geanalyseerd en gebruikt om processen te automatiseren, efficiëntie te optimaliseren en nieuwe diensten te leveren.

IoT heeft het potentieel om verschillende sectoren te transformeren, waaronder gezondheidszorg, landbouw, transport en productie, door de productiviteit, veiligheid en duurzaamheid te verbeteren

## Deelvraag 2 Welke IOT communicatie protocollen zijn er?

Er zijn verschillende IOT communicatie protocollen, de meest gebruikte zijn:

* MQTT (Message Queuing Telemetry Transport), het MQTT protocol is gemaakt voor apparaten die maar een beperkte hoeveelheid resources hebben. Doordat het protocol maar weinig resources vereist is het geschikt voor bijvoorbeeld sensoren en actuatoren. Het protocol maakt gebruik van een publish-subscribe-model, waarbij een apparaat (publisher) een bericht verzendt naar een server (broker), die het bericht vervolgens doorstuurt naar andere apparaten (subscribers) die geïnteresseerd zijn in het ontvangen van dat specifieke bericht. MQTT maakt gebruik van het TCP transport protocol.
* CoAP (Constrained Application Protocol), het CoAP protocol is ook een licht protocol geschikt voor apparaten die niet erg krachtig zijn. CoAP is een protocol dat gebaseerd is op HTTP protocol en kan daarom als alternatief worden gebruikt voor IOT gerelateerde toepassingen. Het protocol maakt gebruik van een client-servermodel, waarbij een apparaat (client) een verzoek indient bij een server om een bepaalde actie uit te voeren of om informatie op te halen. CoAP maakt gebruik van het UDP transport protocol.
* AMQP (Advanced Message Queuing Protocol), is een open source protocol voor berichtensystemen dat is ontworpen voor het versturing van berichten tussen verschillende systemen en applicaties. Het protocol is gemaakt voor betrouwbaarheid, veiligheid en flexibiliteit, maar ook om een manier te hebben hoe systemen zonder problemen met elkaar samen kunnen werken. Het AMQP protocol maakt gebruik van de zogeheten “Producer”, “Consumer” en “Message broker”.

## Deelvraag 3 Hoe pas je het MQTT protocol toe?

Zoals eerder vermeld maakt het MQTT protocol gebruik van het publish-subscribe model. Hierbij moet je een Publisher hebben (Het apparaat dat de berichten verstuurt), een broker (de server die de berichten ontvangt en verwerkt) en een subscriber (degene die zich abonneert op het onderwerp en de berichten ontvangt). De communicatie gaat dan als volgt:

1. Als eerste maakt de publisher verbinding met de broker.
2. De publisher publiceert het bericht naar de broker, hierin wordt ook het onderwerp mee gegeven.
3. De broker onvangt het bericht en voert verschillende controles aan de hand van ingestelde regels
4. De broker stuurt het bericht naar de subscribers die geabonneerd zijn op het onderwerp.
5. De subscriber ontvangt het bericht

MQTT wordt gebruikt op veel verschillende gebieden zoals bijvoorbeeld de logistiek, olie en gas winning en smart homes.

## Deelvraag 4 Hoe pas je het CoAP protocol toe?

Het CoAP protocol maakt gebruik van het client-server model. Hierbij kan de client aan de hand van methodes (GET, PUT, POST, DELETE) en een URI (Uniform Resource Identifier) specificeren wat hij van de server wilt. De communicatie tussen client en server ziet er zo uit:

1. Als eerste maakt de client verbinding met een server, de client gebruikt hierbij het IP-adres van de server en de server luister op een bepaalde poort.
2. De client verstuurt een verzoek naar de server. In dit verzoek wordt een methode meegegeven en een URI. Deze data heeft de server nodig om precies te weten wat de client wil hebben.
3. De server ontvangt het verzoek en controleert of het voldoet aan de geldende beveiligings- en toegangsregels.
4. De server stuurt een antwoord terug naar de client met de gevraagde informatie. Er wordt ook nog een statuscode meegegeven om aan te geven of het gelukt is.

## Deelvraag 5 Hoe werkt het AMQP protocol?

Het AMQP protocol heeft altijd een “message broker” aka server, het centrale stuk, die de berichten van de “producer” ontvangt en doorstuurt naar de correcte “consumer”. De communicatie tussen deze onderdelen zou er als volgt uit kunnen zien:

1. Als eerste moet de "producer" een bericht sturen naar de Message broker/Server.
2. De server Message broker ontvangt vervolgens dit bericht
3. De message broker slaat het bericht op en kijkt welke consumer het moet ontvangen.
4. De message broker stuur het bericht naar de juiste consumer
5. Vervolgens ontvangt de consumer het bericht en verwerkt het verder
6. Op het moment dat het bericht succesvol verwerkt is stuurt de consumer nog een bericht terug naar de broker.

## Deelvraag 6 Wat zijn de verschillen tussen de onderzochte protocollen?

* MQTT is gebaseerd op een publish subscribe model en is een simpeler maar makkelijk schaalbaar protocol. Door het lichtgewicht van het protocol is het makkelijk om te implementeren op apparaten met beperkte resources. MQTT ondersteunt alleen het TCP protocol.
* AMPQ is wat complexer dan MQTT maar in ruil daarvoor heb je een betrouwbaarder en over het algemeen veiliger protocol. Hiernaast ondersteunt AMPQ ook meer protocollen en communicatiepatronen.
* CoAp werkt met een request response model. CoAP is beter geschikt voor toepassingen die een snelle respons vereisen het maakt namelijk gebruik van het UDP protocol wat sneller maar minder betrouwbaar is dan TCP.

## Deelvraag 7 Hoe ziet het smart home systeem eruit?

Het smart home systeem bestaat uit verschillende onderdelen.

* Temperatuur sensoren voor het meten van de temperatuur in de kamer
* Beweging sensoren om het kijken of de bewoner aanwezig is
* Een LDR (Licht sensor) om te kijken of de kamer al verlicht is
* De verlichting van de kamer
* De thermostaat die de temperatuur reguleert

Het systeem gaat dan als volgt in werking in bijvoorbeeld de woonkamer: Op het moment dat er een persoon wordt gedetecteerd in de ruimte gaat het licht aan mits de kamer niet al goed verlicht is door bijvoorbeeld daglicht. Hiernaast wordt ook de temperatuur op een fijne stand gezet zodat het aangenaam is om in de kamer te zijn. Op het moment dat de bewoner de kamer weer verlaat zal de verlichting weer uit gaan en de temperatuur verlaagt worden, om energie te besparen.

## Deelvraag 8 Welk protocol kan het best gebruikt worden bij een smart-home temperatuur regeling?

Voor een smart home systeem met temperatuur regeling is het MQTT protocol erg toepasselijk. Het MQTT protocol is erg lichtgewicht wat ideaal is voor de kleine sensoren en actuatoren met weinig kracht. AMQP is daarom wat minder handig om te gebruiken in deze situatie. Naast dat het protocol goed voor lichtgewicht situaties is, is het ook heel goed schaalbaar. In een smart home setting is dat belangrijk voor als de gebruiker bijvoorbeeld een extra kamer wilt toevoegen aan het systeem. Hoewel het CoAp protocol sneller is, is het wel minder betrouwbaar en dat willen we in een smart home setting niet. MQTT is ook beter geschikt voor het verwerken van grote hoeveelheden data wat in een smart home setting relevant is.

# POC

Om de werking van het systeem in de praktijk te testen maak ik een kleine POC. Deze POC zal een deel van het concept testen. De POC bestaat uit een beweging sensor (Publisher) en LDR (Publisher) op een arduino wifi en een lampje (Subscriber) op een arduino wifi. Als broker gebruik ik mosquitto (<https://test.mosquitto.org/>). Dit systeem zal een deel van de functionaliteit testen die in een kamer geïnstalleerd zou kunnen worden.

Het systeem gaat als volgt te werk: De beweging sensor controleert steeds op een beweging van de gebruiker. Bij een detectie zal hij dit aan de broker doorgeven. Vervolgens zal het lampje dan aan gaan mits de LDR niet detecteert dat er al voldoende licht is.

Afbeelding met tekst, schermopname, Lettertype, software

Automatisch gegenereerde beschrijving

Afbeelding met tekst, elektronica, schermopname, software

Automatisch gegenereerde beschrijving

Afbeelding met tekst, elektronica, schermopname, software

Automatisch gegenereerde beschrijving

# Conclusie

# Bronnenlijst

IOT Handbook van postscapes

<https://www.postscapes.com/iot/>

Guide over IOT protocols van kelton:

<https://www.kellton.com/kellton-tech-blog/your-complete-guide-to-iot-protocols-and-Standards-2022>

ChatGPT voor extra uitleg over de protocollen.

<https://chat.openai.com>

Guide over het MQTT protocol.

<http://www.steves-internet-guide.com/mqtt/>

<http://www.steves-internet-guide.com/mqtt-works/>

What is Amqp en hoe werkt het?

<https://www.cloudamqp.com/blog/what-is-amqp-and-why-is-it-used-in-rabbitmq.html>

MQTT vs AMQP

<https://www.hivemq.com/blog/mqtt-vs-amqp-for-iot/>

Tutorial van mqtt opzetten met arduino wifi:

<https://docs.arduino.cc/tutorials/uno-wifi-rev2/uno-wifi-r2-mqtt-device-to-device>

Mqtt broker mosquitto

<https://test.mosquitto.org/>

Datasheet HC-SR501 bewegingsensor

<https://www.makerguides.com/wp-content/uploads/2019/07/HC-SR501-Datasheet.pdf>