MQTT tehnoloogia ja selle seos raadiosignaali tugevuse soojuskaardi genereerimisega

MQTT (Message Queuing Telemetry Transport) on kerge sõnumsideprotokoll, mis on loodud asjade Interneti (Internet of Things) seadmete ja rakenduste jaoks. See on avatud lähtekoodiga protokoll, mis võimaldab piiratud ressurssidega asjade Interneti-seadmetel saata või avaldada teavet konkreetse teema kohta serverisse, mis toimib MQTT-sõnumivahendajana. Seejärel edastab maakler teabe neile klientidele, kes on teemaga varem liitunud. MQTT on hea valik traadita võrkude jaoks, mille latentsusaeg on erinev tänu oma väikestele sõnumipäistele, mis optimeerivad võrgu ribalaiust[1].

Raadiosignaali tugevuse soojuskaardi genereerimise kontekstis saab MQTT-d kasutada garanteeritud sõnumiedastusega andmete edastamiseks, et anda täpset teavet konkreetse piirkonna signaali tugevuse kohta. Antennilt, GPS-ilt ja raadiosignaali tugevusmõõturilt kogutud andmeid saab töödelda ja visualiseerida soojuskaardina, kasutades MQTT liideseid reaalajas värskendamiseks ja visualiseerimiseks. MQTT võimaldab sõnumeid seadmest pilve ja pilvest seadmesse, muutes sõnumite edastamise asjade rühmadele lihtsaks. See funktsioon on kasulik raadiosignaali tugevuse soojuskaartide loomiseks reaalajas[1].

MQTT pakub sõnumite edastamiseks kolme erinevat teenusekvaliteedi (Quality of Service) taset: 'kõige rohkem üks kord', 'vähemalt üks kord' ja 'täpselt üks kord'. QoS-i tase määrab sõnumi edastamise usaldusväärsuse, mis on oluline paljude asjade Interneti kasutusjuhtude puhul. MQTT püsivate seansside tugi vähendab kliendi ja maakleriga taasühendamise aega, muutes selle sobivaks asjade Interneti-seadmete jaoks, mis ühendavad ebausaldusväärsete mobiilsidevõrkude kaudu[3].

Rakendamise osas on MQTT kliendid väga väikesed ja nõuavad minimaalseid ressursse, mistõttu sobivad need väikestele mikrokontrolleritele nagu käesolevas uurimistöös kasutatud Raspberry Pi seade. MQTT muudab sõnumite krüptimise TLS-i abil lihtsaks ja klientide autentimise kaasaegsete autentimisprotokollide (nt OAuth) abil, tagades edastatavate andmete turvalisuse[3].

Kokkuvõtteks võib öelda, et MQTT on kerge sõnumsideprotokoll, mis sobib ideaalselt asjade Interneti-seadmete ja rakenduste jaoks. See võimaldab piiratud ressurssidega asjade Interneti-seadmetel saata või avaldada teavet konkreetse teema kohta serverisse, mis toimib MQTT-sõnumivahendajana. MQTT-d saab kasutada raadiosignaali tugevuse soojuskaardi genereerimisel, et edastada andmeid garanteeritud sõnumiedastusega ja anda täpset teavet konkreetse piirkonna signaali tugevuse kohta. MQTT püsivate seansside ja turvafunktsioonide tugi muudab selle sobivaks asjade Interneti-seadmete jaoks, mis loovad ühenduse ebausaldusväärsete mobiilsidevõrkude kaudu.

Tsitaadid:

[1] https://www.techtarget.com/iotagenda/definition/MQTT-MQ-Telemetry-Transport

[2] https://www.redbooks.ibm.com/redbooks/pdfs/sg248054.pdf

[3] https://mqtt.org

[4] https://dl.acm.org/doi/10.1145/3529978

[5] https://www.mdpi.com/2076-3417/11/11/4879

MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol that is ideal for IoT devices and applications. MQTT brokers act as intermediaries between MQTT clients, enabling them to communicate with each other. In the context of radio signal strength heat map generation, MQTT can be used to transmit data with guaranteed message delivery to provide accurate information about the signal strength of a specific area.

MQTT maakleri analüüs

1. EMQ X Broker:

Väga skaleeritav ja hajutatud MQTT maakler, mis suudab hallata miljoneid samaaegseid ühendusi. See toetab MQTT 5.0 ja 3.1.1 protokolle ning pakub täiustatud funktsioone, nagu sõnumite marsruutimine, sõnumite filtreerimine ja sõnumite püsivus. EMQ X Broker on avatud lähtekoodiga ja seda saab juurutada kohapeal või pilves. See sobib suuremahuliste IoT-rakenduste jaoks ja suudab käsitleda suurt hulka andmemahtusid erinevatest seadmetest.

2. Mosquitto:

Avatud lähtekoodiga MQTT maakler, mis on kerge ja hõlpsasti kasutatav. See toetab MQTT 5.0 ja 3.1.1 protokolle ning pakub selliseid funktsioone nagu sõnumite püsivus, sõnumite järjekord ja sõnumite filtreerimine. Mosquitto sobib väikesemahuliseks juurutamiseks ja seda saab juurutada kohapeal või pilves. See sobib ideaalselt väikesemahuliste IoT-rakenduste jaoks ning pakub kiiret sõnumiedastust ja vähest mälumahtu.

3. HiveMQ:

Väga skaleeritav ja hajutatud MQTT maakler, mis suudab hallata miljoneid samaaegseid ühendusi. See toetab MQTT 5.0 ja 3.1.1 protokolle ning pakub täiustatud funktsioone, nagu sõnumite marsruutimine, sõnumite filtreerimine ja sõnumite püsivus. HiveMQ sobib suuremahuliste IoT rakenduste jaoks ja suudab käsitleda suure hulga andmemahtusid erinevatest seadmetest. See pakub tugevat jõudlust ja töökindlust, muutes selle sobivaks missioonikriitiliste rakenduste jaoks.

4. RabbitMQ:

Sõnumivahendaja, mis toetab mitut sõnumsideprotokolli, sealhulgas MQTT-d. See pakub selliseid funktsioone nagu sõnumite järjekord, sõnumite suunamine ja sõnumi püsivus. RabbitMQ sobib suuremahuliseks juurutamiseks ja seda saab juurutada pilves. See pakub täiustatud funktsioone, nagu rühmitamine, jaotamine ja tõrkesiirde, muutes selle sobivaks kõrge saadaolevate rakenduste jaoks.

5. AWS IoT Core:

Täielikult hallatav MQTT maakler, mis pakub täiustatud funktsioone, nagu sõnumite marsruutimine, sõnumite filtreerimine ja sõnumite püsivus. See toetab MQTT 5.0 ja 3.1.1 protokolle ning seda saab integreerida teiste AWS-teenustega, nagu AWS Lambda ja AWS IoT Analytics. AWS IoT Core sobib suuremahuliseks juurutamiseks ja seda saab juurutada pilves. See pakub tugevaid turbe- ja mastaapsuse funktsioone, muutes selle ideaalseks ettevõtte tasemel IoT rakenduste jaoks.

6. IBM MQ:

Sõnumivahendaja, mis toetab MQTT-d ja muid sõnumiprotokolle. See pakub selliseid funktsioone nagu sõnumite järjekord, sõnumite suunamine ja sõnumi püsivus. IBM MQ sobib väikesemahuliseks juurutamiseks ja seda saab juurutada kohapeal või pilves. See pakub tugevaid turva- ja töökindlusfunktsioone, mistõttu sobib see missioonikriitiliste rakenduste jaoks.

7. Apache Kafka:

Avatud lähtekoodiga sõnumite vahendaja, mis toetab MQTT-d ja muid sõnumsideprotokolle. See pakub selliseid funktsioone nagu sõnumite järjekord, sõnumite suunamine ja sõnumi püsivus. Apache Kafka sobib suuremahuliseks juurutamiseks ja seda saab juurutada kohapeal või pilves. See on tuntud oma suure läbilaskevõime ja madala latentsusajaga võimaluste poolest, mistõttu on see ideaalne suurte andmemahtude haldamiseks erinevatest seadmetest.

Kokkuvõtteks võib öelda, et saadaval on mitu MQTT maaklerit, mida saab selles uurimistöös tõhusalt kasutada. MQTT maakleri valik sõltub projekti spetsiifilistest nõuetest, nagu mastaapsus, töökindlus ja turvalisus. EMQ X Broker, Mosquitto, HiveMQ, RabbitMQ, AWS IoT Core, IBM MQ ja Apache Kafka on mõned populaarsed MQTT maaklerid, mida võib selle uurimistöö jaoks kaaluda.

Tsitaadid:

[1] https://www.hps.cam.ac.uk/students/partiii-guide/essays-dissertation

[2] https://writingmetier.com/article/ia-word-count-and-structure/

[3] https://community.qualtrics.com/custom-code-12/minimum-word-count-5649?postid=17238

[4] https://chs.harvard.edu/primary-source/hesiod-works-and-days-sb/

MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol designed for IoT (Internet of Things) devices and applications. It is an open-source protocol that enables resource-constrained IoT devices to send or publish information about a specific topic to a server that acts as an MQTT message broker. The broker then pushes the information out to those clients that have subscribed previously to the topic. MQTT is a good choice for wireless networks that experience varying levels of latency due to its small message headers that optimize network bandwidth[1].

In the context of radio signal strength heat map generation, MQTT can be used to transmit data with guaranteed message delivery to provide accurate information about the signal strength of a specific area. The data collected from the antennae, GPS, and radio signal strength meter can be processed and visualized as a heat map using MQTT interfaces for real-time updates and visualization. MQTT allows for messaging between device to cloud and cloud to device, making it easy to broadcast messages to groups of things. This feature is useful for generating heat maps of radio signal strength in real-time[1].

MQTT offers three different QoS (Quality of Service) levels for message delivery: "at most once," "at least once," and "exactly once." The QoS level determines the reliability of message delivery, which is important for many IoT use cases. MQTT's support for persistent sessions reduces the time to reconnect the client with the broker, making it suitable for IoT devices that connect over unreliable cellular networks[3].

In terms of implementation, MQTT clients are very small and require minimal resources, making them suitable for small microcontrollers like the Raspberry Pi device used in this research work. MQTT makes it easy to encrypt messages using TLS and authenticate clients using modern authentication protocols, such as OAuth, ensuring the security of the data transmitted[3].

In conclusion, MQTT is a lightweight messaging protocol that is ideal for IoT devices and applications. It enables resource-constrained IoT devices to send or publish information about a specific topic to a server that acts as an MQTT message broker. MQTT can be used in radio signal strength heat map generation to transmit data with guaranteed message delivery and provide accurate information about the signal strength of a specific area. MQTT's support for persistent sessions and security features makes it suitable for IoT devices that connect over unreliable cellular networks.

[1] https://www.opc-router.com/what-is-mqtt/

[2] https://dl.acm.org/doi/10.1145/3529978

[3] https://www.emqx.com/en/blog/the-ultimate-guide-to-mqtt-broker-comparison

[4] https://www.mdpi.com/2076-3417/11/11/4879

[5] https://www.mdpi.com/1996-1073/14/18/5817