

MQTT Client (Message Queue Telemetry Transport)

Podporované typy a verzie zariadení
Konfigurácia komunikačnej linky
Parametre protokolu linky
Konfigurácia komunikačnej stanice
Konfigurácia meraných bodov
Literatúra
Revízie dokumentu

Podporované typy a verzie zariadení

Protokol je implementáciou štandardu MQTT 3.1.1 (október 2014). MQTT protokol je klient/server protokol typu subscribe/publish. Je jednoduchý, má malú réžiu a je ľahko implementovateľný. Používa sa na komunikáciu M2M (Machine to Machine) a v kontexte IoT (Internet of Things). D2000 KOM implementuje klientskú časť protokolu. Protokol je implementovaný na TCP/IP linke. Pre prenos LoRaWAN dát v rámci MQTT protokolu pozri popis protokolu [LoRaWAN](#).

Komunikácia bola testovaná/nasadená vo:

- cloudu TheThings.Network
- cloudu [Loriot.io](#)
- cloudu Slovanet
- Pixii [PowerShaper](#) (energy storage solution)
- cloudu [liveobjects.orange-business.com](#)

Pozn: bola otestovaná aj komunikácia vo cloudu [liveobjects.orange-business.com](#) cez websockety (<wss://liveobjects.orange-business.com:443/mqtt>). Ako WSS wrapper bol použitý program <https://github.com/jimparis/unwebsockify.git>. Tento bol spustený s parametrami: `./unwebsockify.py --port 1883 --listen 172.16.0.1 wss://liveobjects.orange-business.com:443/mqtt`. D2000 KOM proces sa pripájal k adrese 172.16.0.1 na port 1883. WSS wrapper sa pripojil na definovanú URL a zabalil dáta MQTT komunikácie do websocket obálky.

Každá PUBLISH správa obsahuje tému (Topic), samotné dáta (Payload) a úroveň potvrdzovania (QoS). PUBLISH správy môže odosielať klient aj server. Klient na začiatku komunikácie použije správu SUBSCRIBE na oznámenie, o aké témy (parameter protokolu [Topic Filter](#)) má záujem. Protokol definuje tieto úrovne potvrdzovania PUBLISH správ - QoS (Quality of Service):

- **QoS_0** - správa PUBLISH nie je potvrdzovaná, môže dôjsť ku strate
- **QoS_1** - správa PUBLISH je druhou stranou potvrdzovaná PUBACK, môže dôjsť k duplicite
- **QoS_2** - správa PUBLISH je druhou stranou potvrdzovaná PUBREC, tá je spätne potvrdzovaná správou PUBREL a tá finálne správou PUBCOMP

Úroveň potvrdzovania správ posielaných procesom D2000 KOM definuje parameter protokolu [Publish QoS](#). D2000 KOM proces považuje zápis výstupného meraného bodu za úspešne ukončený v závislosti od QoS:

- **QoS_0** - po úspešnom odoslaní dát cez TCP spojenie
- **QoS_1** - po prijatí PUBACK
- **QoS_2** - po prijatí PUBCOMP

Komunikácia MQTT sa začína správou CONNECT, ktorú pošle klient (D2000 KOM). Správa obsahuje prihlasovacie meno ([User Name](#)), heslo ([Password](#)) a ďalšie parametre, z ktorých sa dá nastaviť [Clean Session Flag](#) a [Client ID](#) (parameter *Will Flag* sa nevyužíva, podobne ako *Will QoS* a *Will Retain*, parameter *Keep Alive* je nastavený na 0). Server odpovedá správou CONNACK s návratovým kódom, ktorý obsahuje informáciu o úspešnosti pripojenia. Následne klient pošle správu SUBSCRIBE s filtrom tém (parameter [Topic Filter](#)), o ktoré má záujem a požadovanou maximálnou úroveň potvrdzovania (parameter [Subscribe QoS](#)).

Server odpovedá návratovým kódom, ktorý obsahuje informáciu o úspešnosti a maximálny QoS, ktorý bol pre požadované témy pridelený. Nasleduje bežná komunikácia, počas ktorej klient aj server posielajú PUBLISH správy (klient s ubovonou témou, server s témami zodpovedajúcimi filteru tém prijatej SUBSCRIBE správy) a podľa hodnoty parametra [QoS](#) prijatých správ PUBLISH ich potvrdzujú. Pokiaľ server nepošle žiadnu správu dlhšie ako [Ping Interval](#) sekúnd, klient pošle PING request správu, na ktorú server musí odpovedať PING response správou (do času špecifikovaného parametrom [Reply Timeout](#)). Pokiaľ dôjde k zmene parametrov na linke, spojenie je zavreté a znovu vytvorené.

Komunikácia bola otestovaná vo MQTT serveru [www.TheThings.network](#).

Konfigurácia komunikačnej linky

- Kategória komunikačnej linky: [TCP/IP-TCP](#).
- Host: IP adresa MQTT servera (prípadne redundantné adresy oddelené iarkou alebo bodkoiarkou).
- Port: číslo portu je štandardne 1883 alebo 8883 pre kryptovanú SSL/TLS variantu.
- číslo linky: nepoužívané, nastavte hodnotu 0.

Pozn: Štandardný port pre MQTT protokol je 1883 resp. 8883 pre SSL/TLS verziu. D2000 KOM neobsahuje implementáciu SSL/TLS varianty protokolu, ale je možné ju nakonfigurovať s použitím utility stunnel <http://www.stunnel.org> pracujúcej v klientskom móde (client = yes). Stunnel bežiaci na rovnakom počítači ako D2000 KOM by mal použiť na lokálnom porte 1883 a po pripojení sa D2000 KOM procesu na tento port by mal komunikáciu zakryptovať pomocou SSL/TLS a poslať na cieľový MQTT server (typicky na port 8883).

Vynútené prerušenie spojenia: Ke sa všetky stanice na linke dostanú do simulácie alebo majú zastavenú komunikáciu, linka sa odpojí (dôjde ku zavretiu komunikovaného socketu). Ke aspoň jedna stanica bude mať vypnutú simuláciu a nebude mať zastavenú komunikáciu (záložka [Parametre](#) objektu Stanica), linka sa znovu pripojí.

Parametre protokolu linky

Dialóg [konfigurácia linky](#) - záložka **Parametre protokolu**.

Ovplyvňujú niektoré voliteľné parametre protokolu. Môžu byť zadané nasledovné parametre protokolu linky:

Tab. . 1

Parameter	Popis	Jednotka / rozmer	Náhradná hodnota
Full Debug	Zapnutie podrobných výpisov o posielaní a prijímaní hodnôt.	YES/NO	NO
User Name	Užívateľské meno použité v správe CONNECT na pripojenie sa k MQTT serveru.	-	
Password	Heslo použité v správe CONNECT na pripojenie sa k MQTT serveru.	-	
Topic Filter	Meno jednej témy alebo filter pre viacero tém posielaný v rámci SUBSCRIBE správy. Filterom špecifikuje MQTT klient témy, v rámci ktorých chce prijímať správy. Pozn: témy sú zoradené hierarchicky, ako oddeľovač slúži lomka (/), ako maska jednej úrovne slúži plus (+), ako maska pre viacero úrovní znak hash (#). Príklady filtra: a/b, level1/+ , # , +/+/+/up Pozn: zmena parametra <i>Topic Filter</i> sa prejaví po reštarte komunikácie - napr. v dôsledku rozpadnutia TCP spojenia, pokiaľ sú všetky stanice na linke vypnuté (StOff) a znovu zapnuté, prípadne po reštarte KOM procesu. V prvých dvoch prípadoch sa posielajú správy UNSUBSCRIBE na pôvodný <i>Topic Filter</i> a následne SUBSCRIBE na nový <i>Topic Filter</i> (to môže byť dôležité pri tzv. perzistentných sessions, keď je zadaný parameter Client ID a MQTT server si pamätá stav klienta aj po rozpade TCP spojenia).	-	#
Subscribe QoS	Požadovaná maximálna úroveň potvrdzovania (QoS) posielaná v rámci SUBSCRIBE správy. MQTT server môže následne poslať PUBLISH správy s takouto alebo nižšou úrovňou potvrdzovania (ale nie vyššou). PUBLISH správy posielané MQTT serverom budú potvrdzované procesom D2000 KOM podľa úrovne potvrdzovania v nich špecifikovanej. Im vyššia je úroveň potvrdzovania, tým viac správ je medzi klientom a serverom vymenených (1 pri QoS_0, 2 pri QoS_1 a 4 pri QoS_2).	QoS_0 QoS_1 QoS_2	QoS_1
Client ID	Unikátny identifikátor klienta (Client Identifier) posielaný v rámci CONNECT správy. Pozn: je možné zadať aj prázdny reťazec - v tom prípade server môže klientovi priradiť unikátnu meno (pokiaľ takúto funkčnosť podporuje) alebo vráti chybu. Pokiaľ nie je zadaný Client ID, bude ale ignorované nastavenie parametra Clean Session Flag (keďže server priradí zákazníkovi unikátnu meno). Testovaný MQTT server (thethings.network) vrátil chybu, pokiaľ bolo Client ID prázdne a Clean Session Flag =NO. Pozn: konkrétny MQTT broker (PIXII.COM) identifikoval klientov iba podľa <i>Client ID</i> . V praxi to spôsobovalo, že dva rôzne D2000 systémy, ktoré sa pripájali k jednému brokerovi, boli považované za jedného klienta a broker pri vytvorení spojenia zrušil existujúce spojenie, ktoré považoval za staré. Po nastavení <i>Client ID</i> na unikátnu hodnotu sa už komunikácia fungovala bez rozpadov spojení.	-	
Clean Session Flag	Parameter Clean Session Flag správy CONNECT. Hodnota No znamená, že server použije aktuálny stav session (spojenia) - napr. po rozpade a znovuoobnovení TCP spojenia. Znamená to, že sú zaslané všetky nepotvrdené PUBLISH správy s QoS_1 a QoS_2 (a voliteľne aj QoS_0 v závislosti na implementácii). Hodnota Yes znamená, že sa session vytvorí nanovo a žiadne nepotvrdené PUBLISH správy nie sú opakované.	YES/NO	NO
Publish QoS	Úroveň potvrdzovania (QoS) použitá pri posielaní PUBLISH správ procesom D2000 KOM. Poslanie PUBLISH správy je dôsledkom zápisu do výstupného meraného bodu s adresou OUT_VALUE . Im vyššia je úroveň potvrdzovania, tým viac správ medzi klientom a serverom poslaných (1 pri QoS_0, 2 pri QoS_1 a 4 pri QoS_2).	QoS_0 QoS_1 QoS_2	QoS_0
Publish Retain	Nastavenie príznaku Retain použité pri posielaní PUBLISH správ procesom D2000 KOM. Aktivácia príznaku Retain spôsobí, že posledná správa poslaná procesom D2000 KOM bude na MQTT serveri dostupná iným klientom okamžite po ich pripojení, ako aj po odpojení D2000 KOM procesu.	YES/NO	NO
Ping Interval	Pokiaľ MQTT server neposlať špecifikovaný časový interval žiadnu správu, D2000 KOM proces pošle výzvu PING request a očakáva odpoveď PING response (do času Reply Timeout). Hodnota 0 posielanie PING request správ vypína. Parameter umožňuje detekovať rozpadnutie TCP spojenia.	sec	60
Payload Type	Nastavenie parsovania správy: <ul style="list-style-type: none"> Text only - správa nie je parsovaná, priradí sa do meraného bodu s adresou IN_TOPIC JSON - správa sa parsuje ako JSON dáta. Ak existuje meraný bod s adresou IN_TOPIC, celá správa bude priradená do neho. Ak existujú body s adresami JA=json_address, budú naplnené príslušnými dátami z JSON správy. Ak v správe také adresy neexistujú, budú merané body zneplatnené. 	Text only JSON	Text only
Time Field Name	Ak Payload Type =JSON, tak názov poľa s časovou značkou. Ak názov nie je zadaný alebo pole nie je nájdené, hodnotám je priradený aktuálny čas. Viac informácií o formáte názvu poľa viď body s adresami JA=json_address .	-	-

Time Mask	Maska pre parovanie hodnoty v poli s asovou znakov. Pozn: od nastavení asových parametrov stanice závisí, i je as interpretovaný ako lokálny alebo UTC s nakonfigurovaným offsetom. Špeciálne masky sú: <ul style="list-style-type: none"> • UNIX - íselná hodnota vyjadruje počet sekúnd od epochy 00:00:00 01.01.1970 UTC. • UNIXMS - íselná hodnota vyjadruje počet milisekúnd od epochy 00:00:00.000 01.01.1970 UTC. 	-	yyyy-mm-dd hh:mi:ss.mss
Ignore Missing Time	Ignorovanie chýbajúcej asovej znaky - pokia sa v JSON payloade nevyskytuje, nevypíše sa varovanie.	YES/NO	NO
Will Flag	Parameter Will Flag správy CONNECT. Hodnota Yes znamená, že server pošle záujemcom správu Last Will v prípade straty spojenia s D2000 KOM procesom.	YES/NO	NO
Will QoS	Úroveň potvrdzovania (QoS) použitá pri posielaní správy Last Will v prípade straty spojenia s D2000 KOM procesom.	QoS_0 QoS_1 QoS_2	QoS_0
Will Retain	Nastavenie príznaku Retain použité pri posielaní správy Last Will v prípade straty spojenia s D2000 KOM procesom.	YES/NO	NO
Will Topic	Topic použitý pri posielaní správy Last Will v prípade straty spojenia s D2000 KOM procesom.	-	
Will Message	Obsah správy Last Will v prípade straty spojenia s D2000 KOM procesom.	-	
Reply Timeout	Pokia do požadovaného času MQTT server neodpovie na správy SUBSCRIBE, UNSUBSCRIBE a PING request, prípadne sa nepodarí nájsť ubovenú správu (a je naitaná iba jej čas), D2000 KOM proces vyhlási chybu, zavrie spojenie a znovu ho otvorí. Hodnota 0 vypína časový limit. Parameter umožňuje reagovať na problematické správanie MQTT servera.	sec	20
Wait Timeout	Timeout akcie pri jednom itaní z TCP spojenia. D2000 KOM opakuje itanie spontánnych dát Max. Wait Retry krát a pokia nenašiel žiadne dáta, je vyhlásený timeout a itanie je ukončené (a môže nasledovať ďalšie itanie alebo prípadne zápis). Zmenšením parametrov Wait Timeout a Max. Wait Retry je možné dosiahnuť rýchlejšiu odozvu D2000 KOM procesu na zápis na úkor vyššej záťaže CPU, pokia MQTT server nemá žiadne dáta. Pozn: ak od MQTT servera prichádza veľa správ a D2000 KOM potrebuje aj zapisovaciu hodnotu, odporúčame nižšiu hodnotu parametra (napr. 0.005 sec), aby zápis nebol blokovaný itaním (v každom prípade ale po 10 prijatých správach nasleduje prerušenie, počas ktorého sa môžu vykonať nahromadené zápisy).	sec	0.100
Max. Wait Retry	Počet opakovaní itania z TCP spojenia. Pozri popis parametra Wait Timeout .	-	3

Konfigurácia komunikanej stanice

- Komunikovaný protokol "**MQTT Client Protocol**".
- Adresa stanice: adresa stanice zodpovedá pou Topic v správe PUBLISH prijatej od MQTT servera. Adresou môže byť konkrétna téma (Topic), regulárny výraz, znak # reprezentujúci všetky témy, alebo topic .* reprezentujúci všetky témy, ktoré nie sú vhodné pre inú stanicu. Priorita spracovania je takáto:
 - Pokiaľ je na linke stanica s adresou #, všetky správy sú smerované na jej určené body a alej sa nehádá.
 - alej sú prehadzované všetky ostatné stanice na linke (s výnimkou adresy *). Ak s Topic zhoduje s adresou stanice, správa je určená pre u a alej sa nehádá.
 - alej sú prehadzované všetky ostatné stanice na linke (s výnimkou adresy *), pričom ich adresa je vyhodnocovaná ako [regulárny výraz](#). Ak s Topic zhoduje s adresou stanice, správa je určená pre u a alej sa nehádá. Stanice sú prehadzované v zostupnom poradí (podľa adresy stanice), takže špecifickejšie výrazy idú ako prvé (napr. *status/battery* pred *status/batt.**)
 - Nakoniec ak existuje stanica s adresou .*, správa je určená pre u.
- Parametre pollingu na záložke *asové parametre* - odporúčaná je hodnota Delay=0.

Konfigurácia meraných bodov

Možné typy hodnôt meraných bodov: **Ci, Co, Txtl, TxtO, Qi, Ci, Co, Ai, Ao, Di, Do, TiR, ToR, TiA, ToA**.

Typ bodu	Adresa	Popis
Body pre itanie dát poslaných MQTT serverom správou PUBLISH. Pozn: hodnoty bodov sú nastavené D2000 KOM procesom v poradí IN_TOPIC , IN_DATA a IN_ID . Nie je nutné, aby konfigurácia obsahovala všetky tri body.		
Txtl	IN_TOPIC	Téma (Topic) prijatej správy PUBLISH.
Txtl	IN_DATA	Dáta (Payload) prijatej správy PUBLISH.
Ci	IN_ID	Identifikátor paketu (Packet Identifier) správy PUBLISH, ktorý závisí od úrovne potvrdzovania (QoS). Pre správy posielané s QoS_0 je identifikátor nulový, pre QoS_1 a QoS_2 je to kladné 16-bitové číslo. Pozn: ak MQTT server posiela aj správy s úrovňou potvrdzovania QoS_0 a je nakonfigurovaný bod ACK_ID , odporúčame v záložke <i>Filter</i> aktivovať v <i>Nová hodnota pri zmene asu</i> , aby opakovaný zápis hodnoty 0 spôsobil generovanie novej hodnoty líšiacej sa iba asovou znakov.
Body pre parovanie JSON správ		

TxtI, TxtO, Qi, Ci, Co, Ai, Ao, Di, Do, TiR, ToR, TiA, ToA	JA=json_ address	<p>Pokia Payload Type=JSON, správa je parsovaná ako JSON dáta. Hodnota <i>json_address</i> udáva názov JSON poa, ktorého hodnota sa má priradiť do meraného bodu.</p> <p>Pre JSON správy, ktoré môžu byť štruktúrované, je podporená syntax <i>level1.level2.level3 ...</i>, napr. <i>rx.current</i> a pokia obsahujú polia (indexované od 1) tak aj <i>level1[index1].level2[index2].level3 ...</i>, napr. <i>rx.gwrx[1].time</i>.</p> <p>Keže samotná JSON správa môže byť pole, adresa môže zaindikať aj indexom, napr. <i>JA=[1].batt_cell_v_avg</i></p> <p>Príklady vi popis meraných bodov typu Envelope protokolu LoRaWAN.</p>
Bod pre potvrdenie prijatia dát MQTT serveru.		
Co	ACK_ID	<p>Ak je definovaný výstupný meraný bod s adresou ACK_ID, D2000 KOM oakáva potvrdenie spracovania každej správy zápisom kópie hodnoty bodu IN_ID. Až následne nastaví do bodov IN_TOPIC, IN_DATA a IN_ID (v tomto poradí) hodnoty z alšej prijatej PUBLISH správy (ak bola medzitým prijatá).</p> <p>V prípade úrovne potvrdzovania QoS_0 je teda nutné opakované zapisovanie do bodu hodnotu 0.</p> <p>Pokia meraný bod ACK_ID neexistuje, hodnoty do bodov IN_TOPIC, IN_DATA a IN_ID sú nastavované ihne po spracovaní PUBLISH správy.</p> <p>Pozn: pre správy prijaté s úrovňou potvrdzovania QoS_0 sa neposiela žiadne potvrdenie MQTT serveru, iba sa zverejnia hodnoty alšej prijatej PUBLISH správy.</p>
<p>Body pre posielanie hodnôt MQTT serveru správou PUBLISH.</p> <p>Pozn: ak má D2000 KOM proces poslať MQTT serveru správy PUBLISH, musia byť definované obidva body v rámci jednej stanice.</p>		
TxtO	OUT_TO PIC	Téma (Topic) v rámci posielanej správy PUBLISH.
TxtO	OUT_VA LUE	<p>Dáta (Payload) v rámci posielanej správy PUBLISH.</p> <p>Pozn: poslanie správy sa uskutoční ako reakcia na zápis do bodu OUT_VALUE (t.j. pokia sa Topic nemení, tak stáť bod OUT_TOPIC nastaví jednorazovo - napr. pomocou štartovacej hodnoty).</p>

Literatúra

Odkazy

Oficiálna stránka MQTT protokolu <http://mqtt.org>

Špecifikácie a štandardy

MQTT 3.1.1 špecifikácia <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html>

ISO/IEC 20922:2016 <http://www.iso.org/standard/69499.html>

Popisy dátových formátov a API

www.loriot.io - Application API Data Format <https://www.loriot.io/home/documentation.html#docu/app-data-format>

www.thethingsnetwork.org - API Reference <https://www.thethingsnetwork.org/docs/applications/mqtt/api.html>



Blog

O protokole MQTT si môžete prečítať blog

- [Komunikácia - MQTT v praxi](#)

Revízie dokumentu

- Ver. 1.0 - 8. august 2017 - vytvorenie dokumentu.
- Ver. 1.1 - 15. október 2021 - podpora parametrov pre LastWill a Retain
- Ver. 1.2 - 27. október 2021 - podpora parsovania JSON správ
- Ver. 1.3 – 1. február 2022 – podpora asových znaiek v správach JSON



Súvisiace stránky:

[Komunikované protokoly](#)