

Architekturintegration

Der MQTT-Broker ist selbst über MQTT an Thingsboard angebunden. Sensordatenpakete werden von Chirpstack aufgenommen und in den Applications mit Tags versehen. Diese Tags werden in Thingsboard verwendet um die Sensordatenpakete Gerätegruppen zuzuweisen. Dies wird im jeweiligen Data-Converter der verbundenen Integration umgesetzt. Ein Gerät kann hierbei mehreren Gruppen zugeordnet werden. Jede Gruppe erhält später ihren eigenen Topic - so können Zugriffe über Abonnements von Kunden/Nutzern gesteuert werden.

[image.png](#)

Um Sensordaten über MQTT an Kunden zu übertragen, werden diese über einen Rulechain-Flow mittels eines eigenen MQTT Clients (bspw. PUB_stg_F&E_01) an TBMQ gesendet. Der Flow ruft hierbei für jedes Datenpaket mehrere Systemattribute ab.

- Die Tenant-ID
- Den Application-Name (aus Chirpstack)
- Die Gruppen-ID

Aus diesen Informationen wird im MQTT-Node der Topic dynamisch generiert. Da dieser an die Kunden ausgespielt wird, sind Tenant und Gruppen durch die IDs (anstatt der Klarnamen) verschleiert. Zur besseren Übersicht werden die Application-Namen im Klartext eingebunden. Ein Topic kann somit beispielsweise so aussehen:

```
7f1857b0-51a9-11f0-8d0d-9b856896f048/df633c20-a8f2-11f0-85bc-9bdb41bca33f/Pumpwerküberwachung/Dragino CS01-LB - Stromüberwachung NSA/#
```

Hierbei wird für jede Gerätegruppe (definiert über Tags in Chirpstack) ein eigener Topic generiert. Wie im obigen Schaubild zusehen, werden einzelne Datenpakete hierdurch auf mehrere Topics dupliziert und können so unterschiedlichen Clients zur Verfügung gestellt werden (Datenkapselung).

Dadurch lassen sich beispielsweise Sensordaten eines Kunden einem anderen zur Verfügung stellen ohne, dass dieser Kunde die übrigen Sensordaten des anderen Kunden einsehen kann.

Die Erstellung der Topics erfolgt somit in Thingsboard vollautomatisch und wird lediglich über Tags in Chirpstack gesteuert.

Wichtig ist hierbei, dass künftig alle Data-Converter so implementiert werden müssen, dass die Tags aus Chirpstack die Sensoren in Gerätegruppen organisieren.

Hier ein Beispiel-Baustein:

[TbLib.js · main · SWHL Digital / IoT / Scripte_Parser / TB Data Converter / Basismodule für Konverter · GitLab](#)

image.png

Rulechain-Flow zur Anreicherung der Datenpakete mit Systemparametern, Topicgenerierung und Übertragung via MQTT-Node.

Der "Publish"-Client zur Verbindung von TB mit TBMQ folgt folgender **Namenskonvention**:

PUB_**[Umgebungskennung]**_**[Tenant]**_**[Indexnummer]**

Beispiel:

- Name: PUB_stg_F&E_01
- Client_ID: PUB_stg_F&E_01
- Passwort: frei wählbar

image.png

Um die Gruppen-IDs zu erhalten ist ein API-call auf das eigene System notwendig. Hierfür ist in jedem Tenant ein eigener User einzurichten, über den auf die API zugegriffen werden kann.

Idealerweise heißt der User in jedem Tenant gleich.

image.png

Revision #1

Created 2026-01-13 06:52:00 UTC by Marcel Sauer

Updated 2026-01-13 12:38:27 UTC by Marcel Sauer