

# MQTT

- [Setup MQTT Integration](#)
- [Variable Substitution](#)
- [Downlinks on MQTT](#)

## Setup MQTT Integration

Navigate to: [Integrations](#) -> [MQTT](#)

Forward data from the Lobar Platform via MQTT Publish to any MQTT broker.

**Broker URL:** must be in one of the following formats:

- `mqtt[s]://host.domain[:port]`
- `tcp[s]://host.domain[:port]`
- `ssl://host.domain[:port]`

Default port for mqtt/tcp is 1883. Default port for mqtt/tcps/ssl is 8883.

**ClientId:** Used on MQTT connect. Default is: `lobaro-{RND}`. {RND} will be replaced by a unique random string.

**Username / Password:** Used to authenticate with the MQTT broker.

**Topic:** The MQTT topic to publish sensor data. (Can use Variable Substitution, for details see below.)

**Listen for Downlinks:** If set the platform will listen to downlinks on topic: `lobaro/downlinks/#`

**Qos Bytes:** Default 0. Can be set to 0, 1 and 2 to set the MQTT QosBytes on publish.

**TLS Client Cert/TLS Client Key:** Set Client Cert and Key with content of cert and key file to use TLS Auth at Broker

**TLS Broker Cert:** Set Broker Cert to validate the Broker based on its Cert.

Example adding MQTT Integration:

## MQTT Integration erstellen

Broker URL\*

ClientId

Username

Password

Topic\*

 **Speichern**

 **Abbrechen**

## Variable Substitution

Some fields offer Variable Substitution that will be substituted with content from the forwarded payload or device metadata on execution of the integration.

Variable substitution is supported at:

- HTTP Integration Target URL
- MQTT Integration Topic name

Variables are enclosed by curly braces:

- e.g. `lobaro/v1/customer/{device.addr}/up` -> `lobaro/v1/customer/70b3d5e050010abc/up`
  - where `{device.addr}` is `70b3d5e050010abc`



The following substitution variables are available:

- Device
  - `device.addr` - Address of the device.
  - `device.name` - Name of the Receiving device inside the platform.
  - `device.serial` - Serial Number of the Receiving Device.
  - `device.id` - Internal id of device. (Should only be used for self integration on the platform.)
- Organisation
  - `organisation.id` - Id of organisation the device is assigned to.
- Data
  - `data.Path.In.Json` - reference any field in the forwarded data

### Example:

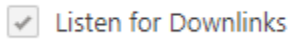
To access the wMbus LinkLayer Id of a telegram use `{data.mbus.IdString}`

You can find the data JSON in the Device Data Tab of the device when expanding the row below "Data (Json)"

RECEIVED	WMBUS RECEIVED
 12.05.2022 05:00:55	12.05.2022 05:00:02
 Data (Json)	
<pre>{   "id": 28943818,   "createdAt": "2022-05-12T03:00:55.832235Z",   "time": "2022-05-12T03:00:02Z",   "type": "parsed",   "datasource": null,   "sensorDataId": "35580174",   "__typename": "ParsedData",   "data": {     "mbus": {       "Id": 1729517622,       "Afl": null,       "Ell": {         "A2Field": {           "Id": 1729517622,</pre>	

## Downlinks on MQTT

To send device Downlinks via MQTT you need to enable the "Listen for Downlinks" feature.



- The platform listens on topic `lobaro/downlinks/#`

### Downlink Topics:

- `lobaro/downlinks/id/<device-id>/<type>`
- `lobaro/downlinks/addr/<device-address>/<type>`

### Placeholders:

- `<device-id>`: Lobar Platform internal device ID of the target device
- `<device-address>`: Address of the target device
- `<type>`: Type of downlink, please refer to [Downlinks](#)

### Body:

The Body of the message needs to be filled according the specifications found in the [Downlinks](#) Documentation.



Lorawan Downlinks need a default LoraWan Server for your org defined under Integrations LoRaWan!