Example of sending M-Bus data to a server

The information in this page is updated in accordance with **00.07.03.1** firmware version .

☐

Contents

Introduction

This article contains instructions on how to send M-Bus data to the server using various protocols.

Configuration overview and prerequisites

Before we begin, let's take a look at the configuration that we are attempting to achieve and the prerequisites that make it possible.

Prerequisites:

- TRB143;
- M-Bus device;
- Server;

- An end device (PC, Laptop, Tablet, Smartphone) for configuration;

If you're having trouble finding any page or some of the parameters described here on your device's WebUI, you should turn on **"Advanced WebUI"** mode. You can do that by **clicking** the **"Basic"** button **under "Mode,"** which is located at the top-right corner of the WebUI.

Node-RED installation and setup

We are going to set up Node-RED in a Linux virtual machine. For Node-RED to work, you would need to install Node.js version 14.00 or higher. If you already have Node.js installed, verify the Node.js version using this command:

```
node -v
```

If you do not have Node.js installed, run these commands to install it:

```
sudo apt install curl
curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash —
sudo apt-get install -y nodejs
```

Once we have Node.js installed, we can install Node-RED. Use this command to install:

```
sudo npm install -g —unsafe-perm node-red
```

Use the command *node-red* to start a local server. Here is how the terminal should look like if the server starts correctly:

Once you have the local server running, use the local IP and port number in your internet browser. In this case, we are using 127.0.0.1:1880:

For MQTT usage, we are going to need MQTT-specific nodes. Use the side menu to navigate to the Manage Palette section and install these nodes:

- node-red-contrib-aedes
- node-red-contrib-mqtt-broker

Configuration

M-Bus Data to Server via MQTT

This section contains information on how to send M-Bus data to the server on Node-RED using the MQTT protocol.

The M-Bus meter is directly connected to TRB143. Gateway reads the M-Bus data and sends it to the server using the MQTT protocol.

Data to Server configuration

The **Data to Server** feature provides you with the possibility to set up data senders that collect data from various sources and periodically send it to remote servers. This section will walk you through configuring the Data to Server to send M-Bus data to the Node-RED server. To configure Data to Server, **access the router's WebUI** and navigate to **Services**, then **Data to Server**:

1. **Enable** instance;
2. **Name:** Enter **desired instance name**;
3. **Data source:** M-bus;
4. **Protocol:** MQTT;
5. **JSON format:** Enter what **data** you would like **to send**;
6. **URL/Host/Connection string:** Enter **address of server**;
7. **Port:** Enter **server port**;
8. **Topic**: Enter **desired topic name**

M-Bus configuration

The **M-Bus Settings** section is used to configure the general service functionality. To set up a new M-Bus instance, go to **Services → M-Bus:**

1. **Enable** M-Bus;
2. **ID**: Enter desired **instance name**;
3. **Add**;

After this, you should be redirected to the configuration page for the M-Bus instance:

1. **Enable** instance;
2. **Name**: Enter your **desired name**;
3. **Period**: Enter **desired period**;

At the bottom of the M-Bus instance, you should see the Request Configuration section. It is used to configure requests from M-Bus devices.

1. **Enable** request configuration;
2. **Slave address:** Enter **desired slave address**;
3. **Data type:** Select desired data type (this time we will be using **HEX**);
4. **Save & apply** changes.

Node-RED configuration

Below you can see the Node-Red block diagram that is used to receive data and its configuration:

- **aedes broker block** - Broker;
- **MQTT in block** - mbus;
- **Debug** - debug 2;

Configure the aedes broker block and broker blocks accordingly:

1. **MQTT Port**: 1883;
2. **WS Bind**: port;
3. **Done**;

1. **Output**: msg. payload;
2. **To**: debug window;
3. **Done**;

Adjust the MQTT IN node:

1. **Edit:** Configure MQTT-Broker node;
2. **Name**: Enter **desired MQTT-Broker nodes name;**
3. **Server**: **127.0.0.1** as we will be using **Node-RED as MQTT-Broker**;
4. **Port**: 1883;
5. **Update**: Save the changes;
6. **Server**: Select **MQTT-Broker** that you have **just created**;
7. **Action**: Subscribe to single topic;
8. **Topic**: Enter the **topic** that you have **set in Data to Server configuration**;
9. **Qos**: 0;
10. **Output**: Auto-Detect;
11. **Done**: Save the changes;

Results

If you have taken all of the steps described above, the configuration is complete. Below you can see M-Bus data output in HEX format.

M-Bus Data to Server via HTTP

This section contains information on how to send M-Bus data to the server on Node-RED using the HTTP(S) protocol.

The M-Bus meter is directly connected to TRB143. Gateway reads the M-Bus data and send it to the server using Hthe TTP(S) protocol.

Data to Server configuration

To set up Data to Server, navigate to **Services → Data to Server:**

1. **Enable** instance;
2. **Name:** Enter **desired instance name**;
3. **Data source:** M-bus;
4. **Protocol:** HTTP(S);
5. **JSON format:** Enter what **data** you would like **to send**;
6. **URL/Host/Connection string:** Enter **address of server** and **port**;

M-Bus configuration

Go to Services → **M-Bus** → the **instance** you have created → **edit.** After that, scroll down to **Request configuration → edit.** There, you will be able to change the data type.

1. **Enable** request configuration;
2. **Slave address:** Enter **desired slave address;**
3. **Data type:** Select desired data type (this time we will be using **XML**);
4. **Save & apply** changes.

Node-RED configuration

Below you can see the Node-Red block diagram that is used to receive data and its configuration:

- **TCP in block** - tcp:8080;
- **Debug** - debug;

Configure TCP in and debug blocks accordingly:

1. **Type**: Listen on;
2. **Port**: 8080;
3. **Output**: Stream of String;
4. **Done**;

1. **Output**: msg. payload;
2. **To**: debug window;
3. **Done**;

Results

If you have taken all of the steps described above, the configuration is complete. Below you can see M-Bus data output in XML format.

MQTT broker

This section contains information on how to send M-Bus data to the server on Node-RED using a Teltonika router as a MQTT broker.

The M-Bus meter is directly connected to TRB143. Gateway acts as an MQTT broker; it reads the M-Bus data and sends it to the MQTT subscriber.

Data to Server configuration

To set up Data to Server, navigate to **Services → Data to Server:**

1. **Enable** instance;
2. **Name:** Enter **desired instance name**;
3. **Data source:** M-bus;
4. **Protocol:** MQTT;
5. **JSON format:** Enter what **data** you would like **to send**;
6. **URL/Host/Connection string:** Localhost IP;
7. **Port:** enter **server port**;
8. **Topic**: enter **desired topic name**;

MQTT broker configuration

The **MQTT Broker** is an entity that listens for connections on the specified port and relays received messages to MQTT clients. To set up MQTT Broker go to S**ervices → MQTT → Broker:**

1. **MQTT broker**: On;
2. **Local port**: 1883;

M-Bus configuration

Go to Services → **M-Bus** → the **instance** you have created → **edit.** After that, scroll down to **Request configuration → edit.** There, you will be able to change the data type.

1. **Enable** request configuration;
2. **Slave address**: Enter **desired slave address**;
3. **Data type**: Select desired data type (this time we will be using **BIN**);

4. **Save & apply** changes.

Node-RED configuration

Below you can see the Node-Red block diagram that is used to receive data and its configuration:

- **MQTT in block** - mbus;
- **Debug** - debug;

Configure debug block:

1. **Output**: msg. payload;
2. **To**: debug window;
3. **Done**;

Adjust the MQTT in node:

1. **Edit:** Configure MQTT-Broker node;
2. **Name**: Enter **desired MQTT-Broker nodes name;**
3. **Server**: Enter MQTT Broker IP;
4. **Port**: 1883;
5. **Update**: Save the changes;
6. **Server**: Select **MQTT-Broker** that you have **just created**;
7. **Action**: Subscribe to single topic;
8. **Topic**: Enter the **topic** that you have **set in Data to Server configuration**;
9. **Qos**: 0;
10. **Output**: Auto-Detect;
11. **Done**: Save the changes;

Results

If you have taken all of the steps described above, the configuration is complete. Below you can see M-Bus data output in BIN format.

See Also

- [M-Bus](#)
- [Data to Server](#)
- [MQTT Broker](#)

External links

[Node-RED](#)