https://wiki.teltonika-networks.com/view/Modbus_data_to_server_via_MQTT_TLS

Modbus data to server via MQTT TLS

□

Contents

Introduction

MQTT is a publish-subscribe based protocol messaging protocol designed for connections with remote locations where network bandwidth is limited. In **MQTT**, a secure connection between a server and client is achieved by implementing **TLS** (Transport Layer Security) as it uses a handshake mechanism where various parameters are negotiated. This article provides guideline on how to secure an **MQTT** connection using **TLS**.

Prerequisite

The following prerequisites make it possible to achieve the desired outcome, which is creating a secure **MQTT** connection using **TLS**.

- One RUTxxx device
- A computer for running applications.
- Modbus Server and MQTT subscriber application (for this example **modRSsim2** and **MQTT.fx v1.7.1** applications were used)

Topology

Configuring Modbus TCP/IP Client

For this example, we will use the router as the Modbus TCP Client that will be requesting data from

Modbus TCP Servers

1. Turn on “Advanced WebUI” mode
2. Navigate to **Services → Modbus → Modbus TCP Client**
3. Click **“Add”** to add a new Modbus TCP Client
4. Enable the Modbus TCP Client Configuration and enter the **“Name, Server ID, IP Address, Port”. Note: Server’s ID, IP Address and Port should be the same as configured in the client device.**
5. Click **“Save”** and the slave device will be added.

Testing

To test whether the functionality is working, we can add a Modbus request to obtain data from the Modbus server. The client sends a request to a specifying the function code to be performed. Then the requested data is sent back to the Client. A maximum of 64 request configurations for each device can be created.

1. Click **“Edit”** button in client device’s configuration.
2. Type the name of the request followed by clicking the **“Add”** button in Requests Configuration section to create a new request.
3. Specify the **“data type, function, first register, number of registers”** and ensure that the request is enabled.
4. Click **“Save & Apply”** to save the changes.
5. Click **“Edit”** button
6. Select the request under **“Request Configuration Testing”** and click to **“Test”**.
7. The server device will return the requested data.

Generating TLS Certificate

To secure an MQTT connection using TLS, we need to generate certificates and for this example, the certificates will be generated from the device.

1. Navigate to **System > Administration > Certificates**.
2. Click on “GENERATE”
3. Navigate to the next tab CERTIFICATES MANAGER
4. Download the highlighted certificates: ca.cert.pem, client.cert.pem, ca.key.pem and client.key.pem to your Windows 11 Computer for easy upload to the MQTT client.

Configuring Data to Server

The Data to Server feature provides the ability to set up data senders that collect data from various sources and periodically send it to remote servers. For this example, we are going to follow the steps below.

1. Access the router’s WebUI and navigate to Services > Data to Server.
2. Click “Add” to add a data sender.
3. Select Modbus in the Type area.
4. Select Json in the Format type.
5. Click on the NEXT: COLLECTION EDIT BUTTON
6. Click on enable.
7. Choose the Json format in format type.

8. Click on **ADVANCED SETTINGS**.
9. The settings should be Period:60, turn on the Retry option, retry count:10 Timeout:1 and then click on **Next: Server configuration**.
10. Select type to MQTT and fill in the details as so. Turn on the TLS, and certificates from device and choose the client certificates.
11. Navigate to **Services>MQTT>Broker** and fille in the settings as so and turn on TLS and turn on the certificates file from device.

MQTT Configurations

Configuring MQTT Client

For this example, we are going to use MQTT.fx v1.7.1 application to securely connect to the MQTT broker, in this case, the router, and subscribe to a topic.

1. Open MQTT.fx and click the settings button to create a new profile.
2. Click “+” to create a new profile.
3. Enter the “Profile Name”, “Profile Type”, “Broker Address” and “Broker Port”.
4. Tick the “Enable SSL/TLS” check box under “SSL/TLS” section, ensure that you select the appropriate protocol, select “Self-signed certificates”, upload the certificates, and select the “PEM Formatted” checkbox.
5. Click “Apply” to save the changes and “Ok” to close the configuration window
6. Click “Connect” to establish connection to the MQTT Broker. Upon successful connection the status should change from “grey” to “green”
7. Navigate to the “Subscribe”, enter the topic that you wish to subscribe to and click “Subscribe”. You should see data coming from the server showing that there has been a successful secure MQTT connection using TLS.

See Also

External Links