iTrixx MQTT Gateway & WFMN Bundle Setting Instructions

NOTE: The configuration demonstrated in this document is intended only to validate communication between a client and the iTrixx-GW MQTT Gateway (Raspberry Pi with Mosquitto broker). This environment is not meant to represent a full production environment since no access control or security are detailed below. Please consult a qualified MQTT consultant for best practices regarding setting up a production MQTT environment, or refer to the MQTT documentation linked here: https://mosquitto.org/documentation/

This guide shows how to configure and implement iTrixx MQTT Gateway and to configure Linortek products to publish data to the broker. To see the published messages, Use **Mqtt-spy** on Windows, and **MQTT Client** on Android to confirm functionality.

Setup the Broker

The iTrixx MQTT Gateway is a tiny, dual-display, desktop computer, you will need a monitor, a keyboard and a mouse for initial setup. First, make sure the device is plugged into the included power supply and connected to the network. The process for connecting the device to the network is similar to doing so on a desktop computer. Connect the device to a monitor using the included HDMI cord and connect a mouse and keyboard to the device. Once done, the Gateway will work like a fully functioning Linux computer. You can connect to the network via an Ethernet cable, or WiFi. Also, the Terminal can be found on the bottom left of the screen.

Version 1.0
Date: July 14, 2021

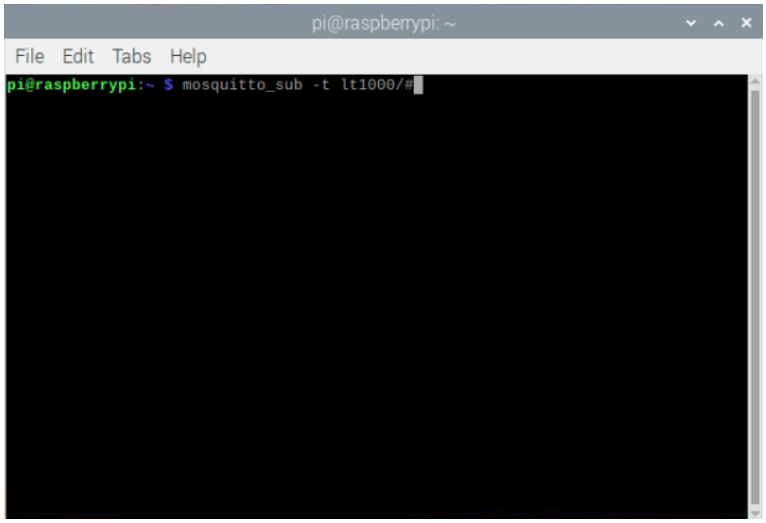
Once connected to the internet, you can proceed to the next steps.

Configuring the Broker

We have already installed the Mosquitto MQTT broker on the Gateway. For some reasons if you need to re-install, type in the following commands into the Terminal:

- sudo apt update.
- sudo apt upgrade
- sudo apt install mosquitto -y
- sudo apt install mosquitto-clients -y

At this time, the WFMN only publishes under the topic: lt1000/xx:xx:xx:xx:xx:xx/tele – where xx:xx:xx:xx:xx:xx is the device's MAC address. It currently sends a single payload in JSON format to the configured broker on a 1-minute interval at QoS 0. To run Mosquitto, click the **Terminal** icon, a window will be opened, enter the command: mosquitto_sub -t lt1000/#.



Your broker is now subscribed to the **lt1000/#** topic.

Setting WFMN to Publish to the Broker

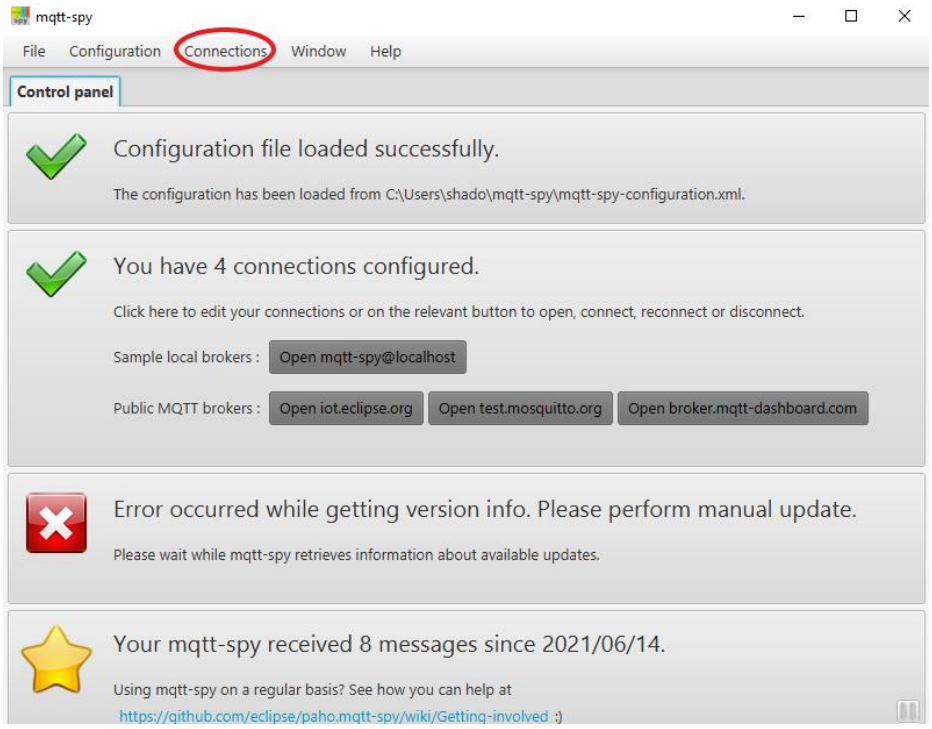
After setting up your broker, you will need to configure the WFMN to connect to the broker. In this example, the WFMN is on the same local network as the broker. The broker's address will be the IP address of the Gateway it is running on. Using telnet, log in to your WFMN and enter the command: mqtthost=brokeraddress. In this case the command is: mqtthost=172.16.1.41. Then set the port by entering the command: mqttport=xxxx which is defaults to 1883. Refer to your broker installation instructions and procedure for details regarding the broker's port number. In this case the command is: mqttport=1883. The WFMN will now publish its payload at a 1-minute interval.

Configuring Clients

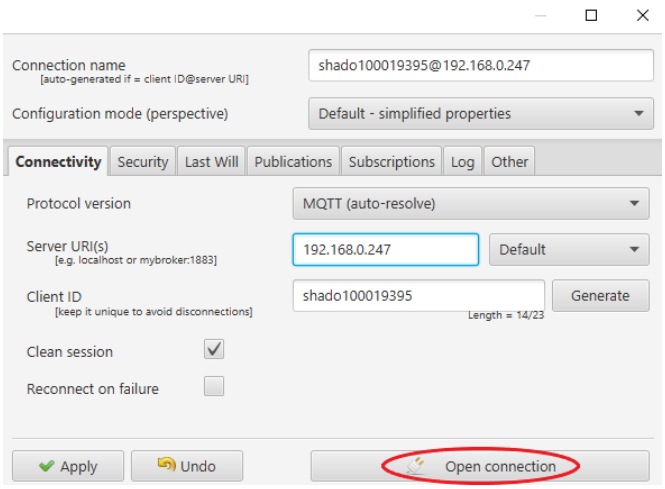
As there are many clients that may now subscribe to **lt1000/#**, this example will use **Mqtt-spy** on Windows and **MQTT Client** on Android.

Mqtt-spy

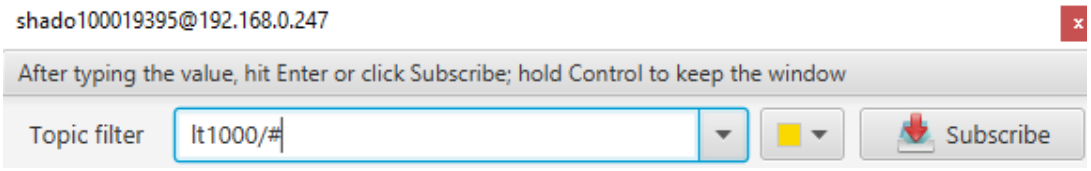
Mqtt-spy is an open-source java application for monitoring MQTT topics. Mqtt-spy is free to download at https://www.eclipse.org/paho/components/mqtt-spy/. After downloading it, open the application and click on the Configuration dropdown menu and select Restore Defaults. Then select "Configure mqtt-spy using sample settings. Then, click the **Connections** dropdown menu and select **New Connection**. A window will open allowing you to configure your connection to the MQTT broker.



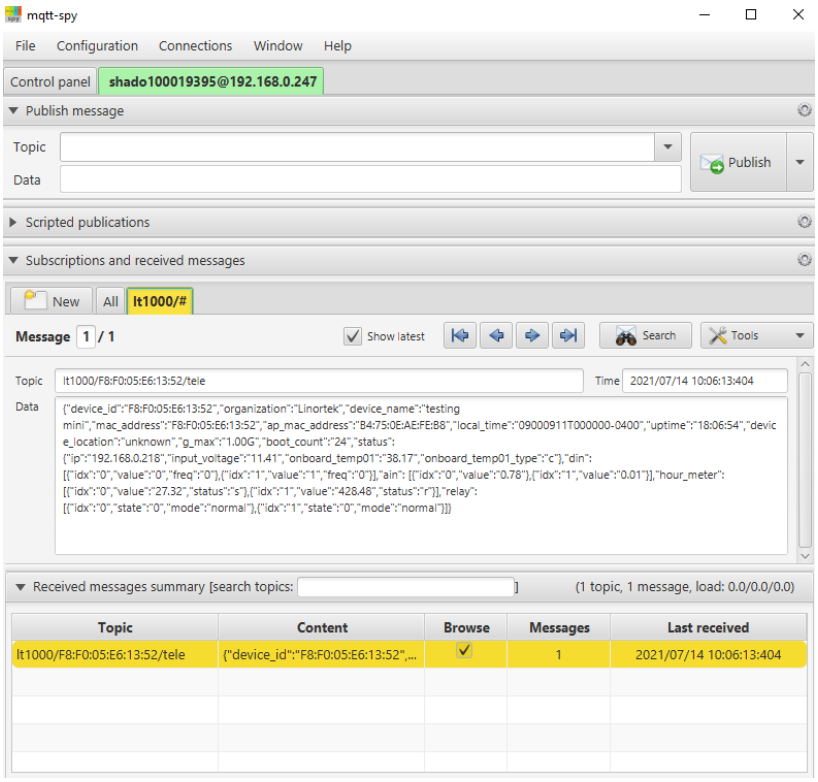
From here, you can set a **Connection Name**, **Server URI**, and **Client ID**. **Server URI** is the address of the server. **Change the Server URI to the Raspberry Pi's IP address.** If you set your broker to require a username and password, this can be entered in the **Security** tab. Once you are finished, click **Open Connection**.



A new tab will open on the main window with your new connection. Click **New** under **Subscriptions and received messages** and enter the topic you wish to subscribe to. In this case there are two Raspberry PIs connected to the broker so to receive data from both, type in **lt1000/#**.



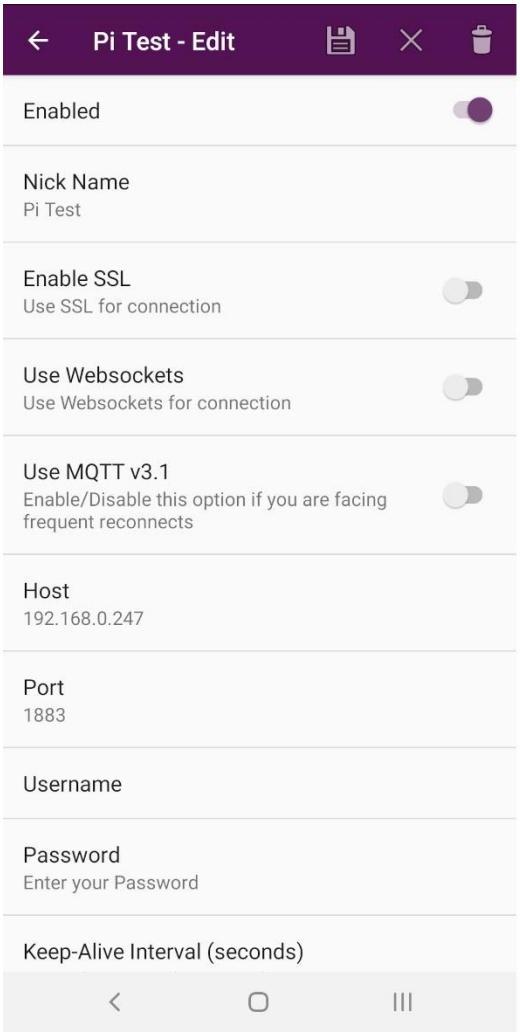
Click **Subscribe** and it will begin receiving data from your WIFI Mini Hour Meter.

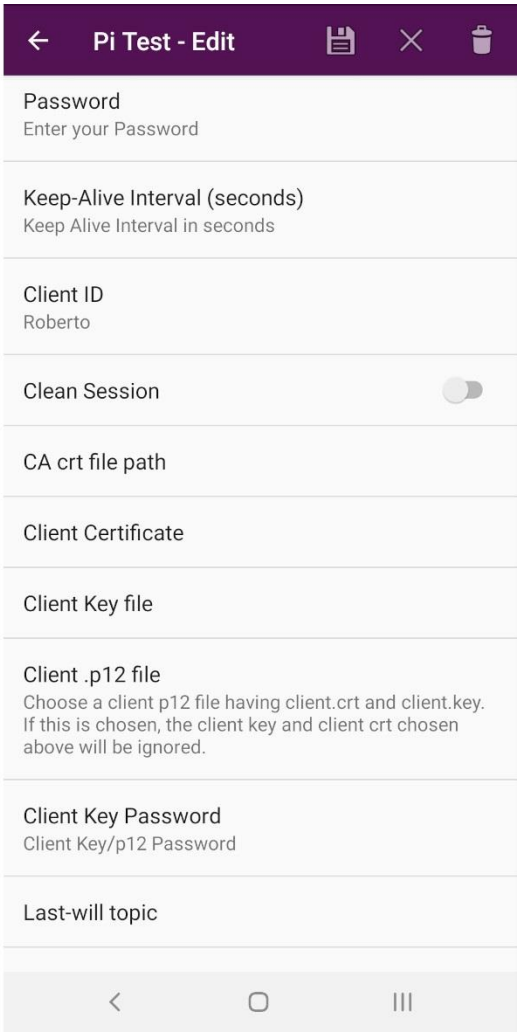


MQTT Client

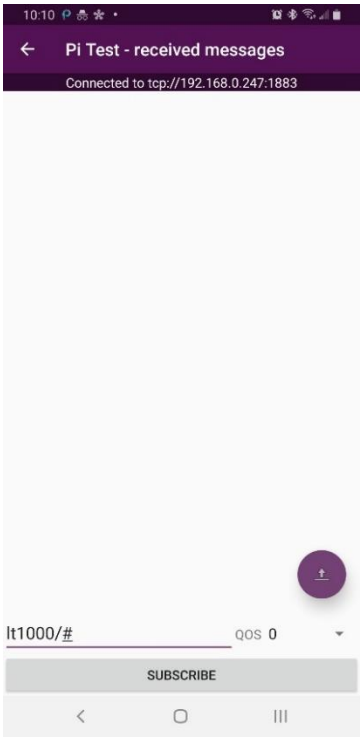
MQTT Client is a free client available for Android from Google Play. Upon first opening the app, you will be presented with a blank screen with a "**+**" sign in the bottom right. Tap to add your broker. On the next screen, tap **Enabled**, assign a **Nick Name**, enter your **Host**'s IP or web address and **Port** number. You may enter a **Username** and **Password** if you have set your broker to require credentials, and enter a **Client ID**. Once complete, tap the **Save** icon in the upper-right and your configured broker will be added to the main screen.

Version 1.0
Date: July 14, 2021





Tap your broker and you will come to a new screen where you can add topics to subscribe to. Tap **Subscribe to a Topic** on the bottom of the screen and enter the topic. In this case, there are two Raspberry PIs connected to the broker, so to receive data from both the topic **lt1000/#** will be used. Once entered, tap **SUBSCRIBE.**



The topic will display on the screen with a preview of the payload contents. Tap the topic to view.

It is now verified that the phone is connected to and communicating with the broker.

Version 1.0
Date: July 14, 2021