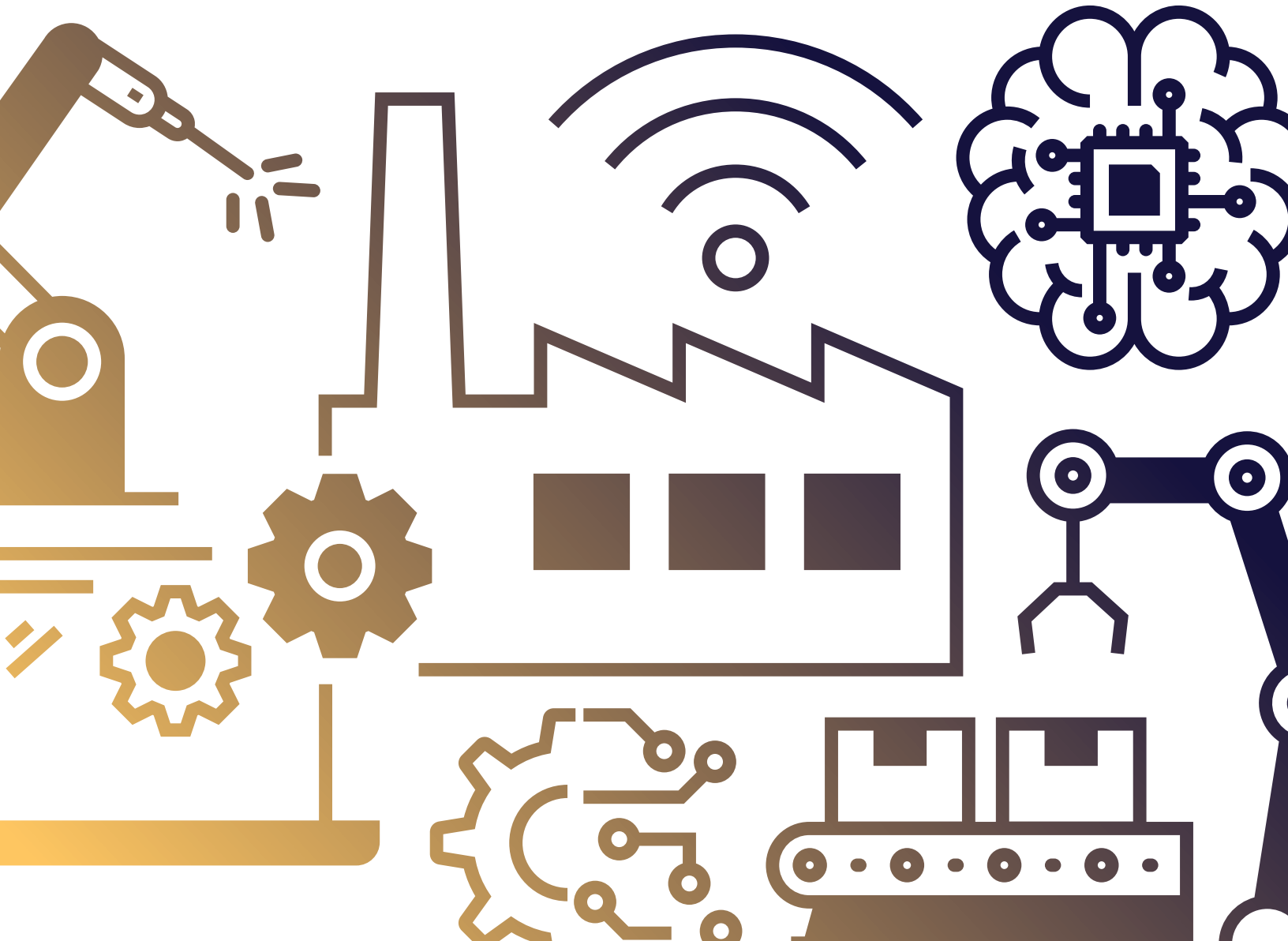
VERSION 8

GAGETRAK MQTT BROKER

Intro Guide



U.S. GOVERNMENT RESTRICTED RIGHTS

This software and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (b)(3)(ii) of The Rights in Technical Data and Computer Software clause at 252.227-7013. Contractor/manufacturer is CyberMetrics Corporation.

TECHNICAL SUPPORT

M-F 6:00 a.m. to 4:00 p.m., MST.
https://gagetrak.com/support/

TRADEMARK ACKNOWLEDGMENTS

All CyberMetrics Corporation products are trademarks or registered trademarks of CyberMetrics Corporation. All other brand and product names are trademarks or registered trademarks of their respective holders.

Entire Contents Copyright 1988-2025, CyberMetrics Corporation All Rights Reserved Worldwide.

GTMQTT8.8-05142025

TABLE OF CONTENTS

Chapter 1: Introduction 4

Chapter 2: Broker Installation 5

Chapter 3: Setting the Root Certificate 6

Chapter 4: Configuration 9

Chapter 5: GAGEtrak Publisher Client 12

Chapter 6: Sample Projects 14

Chapter 7: Conclusion 19

Training 20

Contact Us 20

CHAPTER 1: INTRODUCTION

The GAGEtrak MQTT Broker is a companion product to GAGEtrak Pro/Lite Calibration Management Software. Once configured in your MQTT ecosystem, the GAGEtrak client can publish event-based access to changes made to data within the GAGEtrak application.

When you install the broker on a host machine in your network and configure the GAGEtrak clients to publish data, the Broker can process the information and send it to any subscriber in your MQTT environment.

The GAGEtrak MQTT Broker and the MQTT Publisher component in the GAGEtrak Pro/Lite client were designed to work together. However, the GAGEtrak MQTT Broker can be made to work with other publishers/subscribers in your MQTT ecosystem. The same holds true for the GAGEtrak client; it can function as a stand-alone publisher.

USE CASE

Your GAGEtrak Calibration Management Software should be able to publish data to MQTT topics like:

- Gages/Status
- Calibration/Status
- Calibration/Due Date

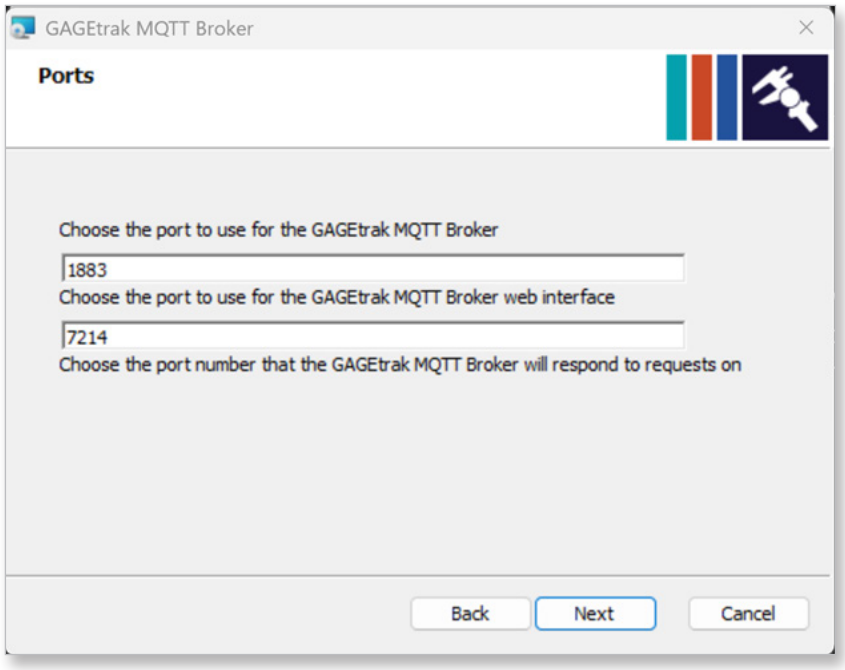
Each time a user updates a gage record; the data is transmitted to the MQTT broker using a standardized payload format such as JSON or URI. Systems subscribed to those topics, such as a statistical process control (SPC) application, ERP or an analytics engine, receive updates in real-time, enabling:

- Instant alerts
- Workflow automation
- Data aggregation for compliance

CHAPTER 2: BROKER INSTALLATION

The GAGEtrak MQTT Broker should be installed on a Windows machine that is always running, such as a server or a hosted VM. The application interface for the GAGEtrak Broker runs in a browser; however, since it runs as its own service, you do not need to install and configure it on a web server, such as IIS or Apache. During installation, you will get the choice of installation location, and after installation, the broker will run as an application service on that machine or VM.

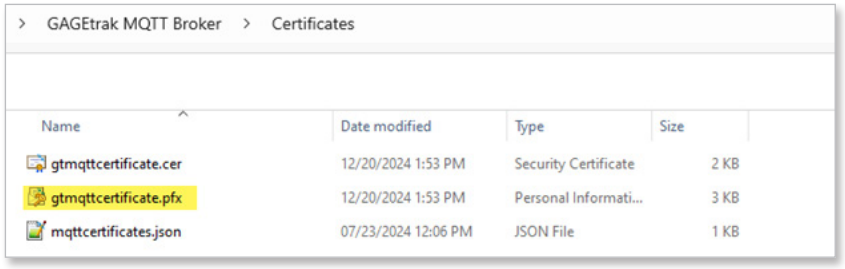
The default ports are unencrypted ports reserved for MQTT communications (these ports are typically used in a local network or VPN). Port 8883 is reserved for MQTT over SSL/TLS (Secure Sockets Layer/Transport Layer Security), ensuring encrypted communication outside of the local network. If you must utilize different ports, be certain they are compatible TCP/IP ports for MQTT communications. Be sure to note the ports that are used, as they will be needed to access the broker and in your code for subscribers and publishers.



CHAPTER 3: SETTING THE ROOT CERTIFICATE

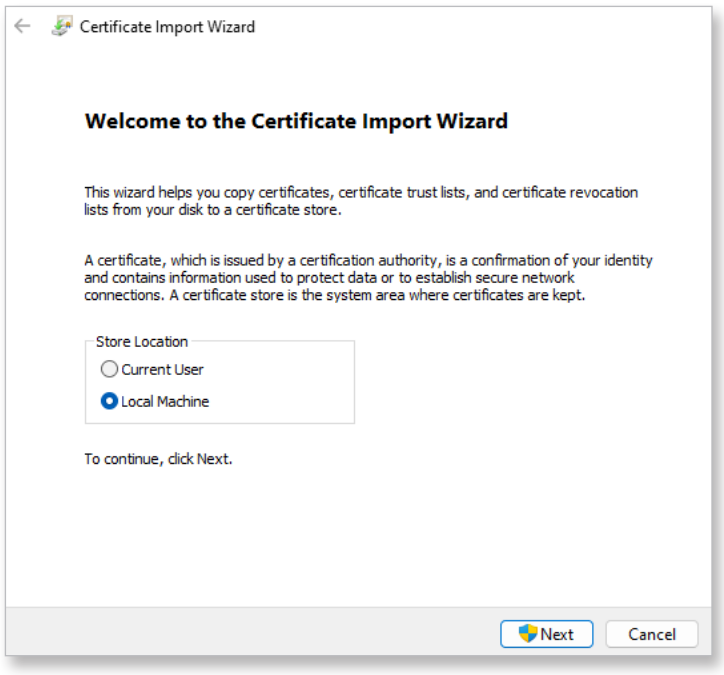
Once the broker is installed, you will need to activate the certificate before you can access the management interface.

Locate the **GAGEtrak MQTT Broker Certificates** folder in the install directory:
C:\Program Files (x86)\CyberMetrics\GAGEtrak MQTT Broker\Certificates.



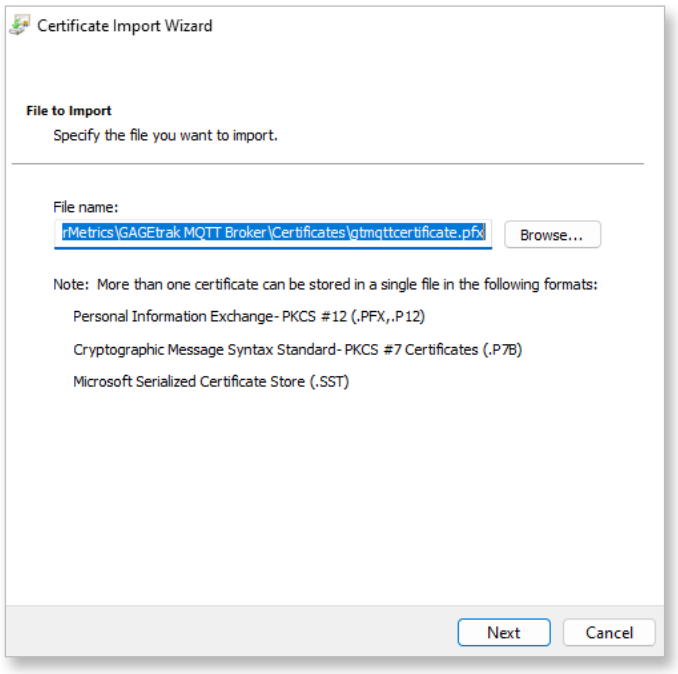
Double-click the **gtmqttcertificate.pfx** file to begin the process of creating a certificate for your MQTT Broker install. **Note:** This is necessary to gain https: access to the GAGEtrak MQTT Broker from your install machine's browser. Once you access the broker, you can replace this cert with your own self-cert or a third-party cert.

On the first dialog, choose **Local Machine** and then click **Next**.

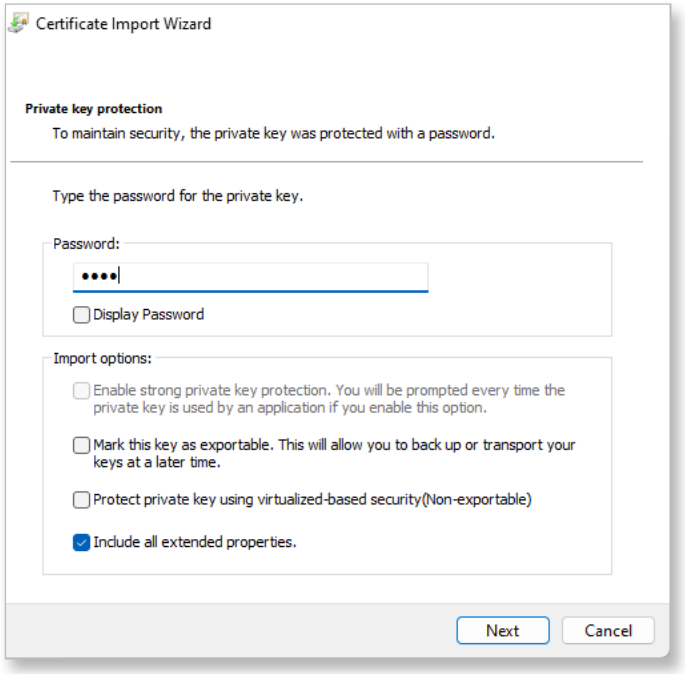


Note: You may be prompted to allow for Rundll; choose **Yes**.

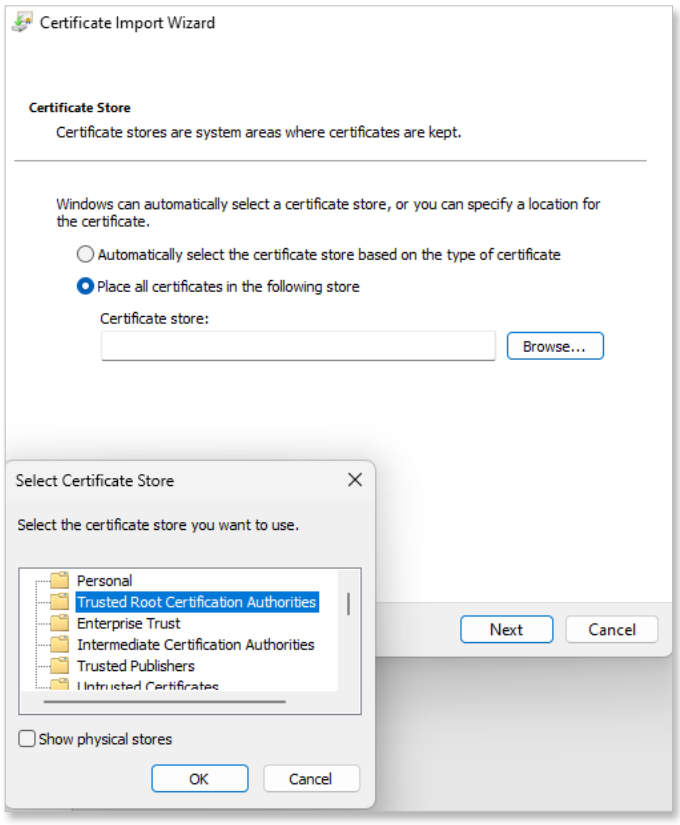
On the **Certificate Import** form, use the default directory provided and click **Next**.



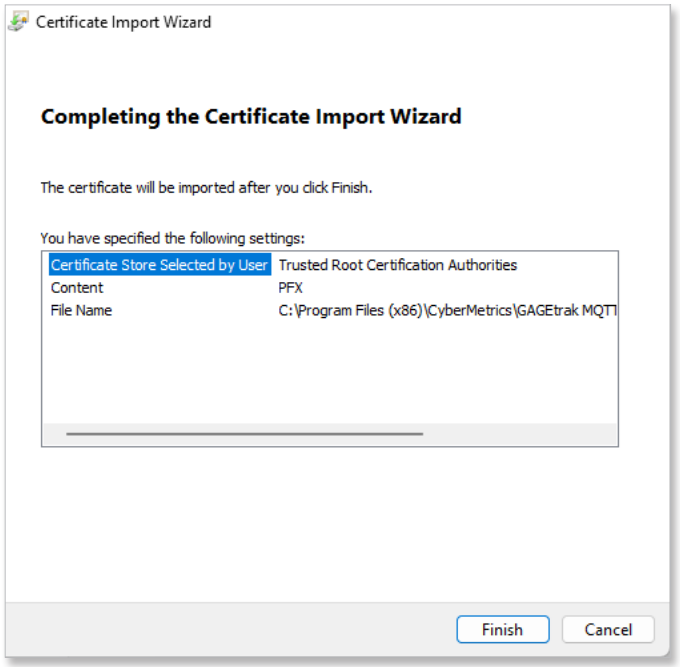
When you see the **Password** form prompt, use *1234*, and leave **Include all extended properties** selected.



On the certificate root selection, choose **Place all certificates in the following store** and then select the **Trusted Root Certification Authorities** folder.



The next form completes the certificate import process. Close this window and proceed to the next step in these instructions.



Open a browser application on your GAGEtrak MQTT Broker install machine and enter ***https://localhost:7214*** in the URL locator. Select the **License** menu item on the left and enter your **License Code** for the broker.

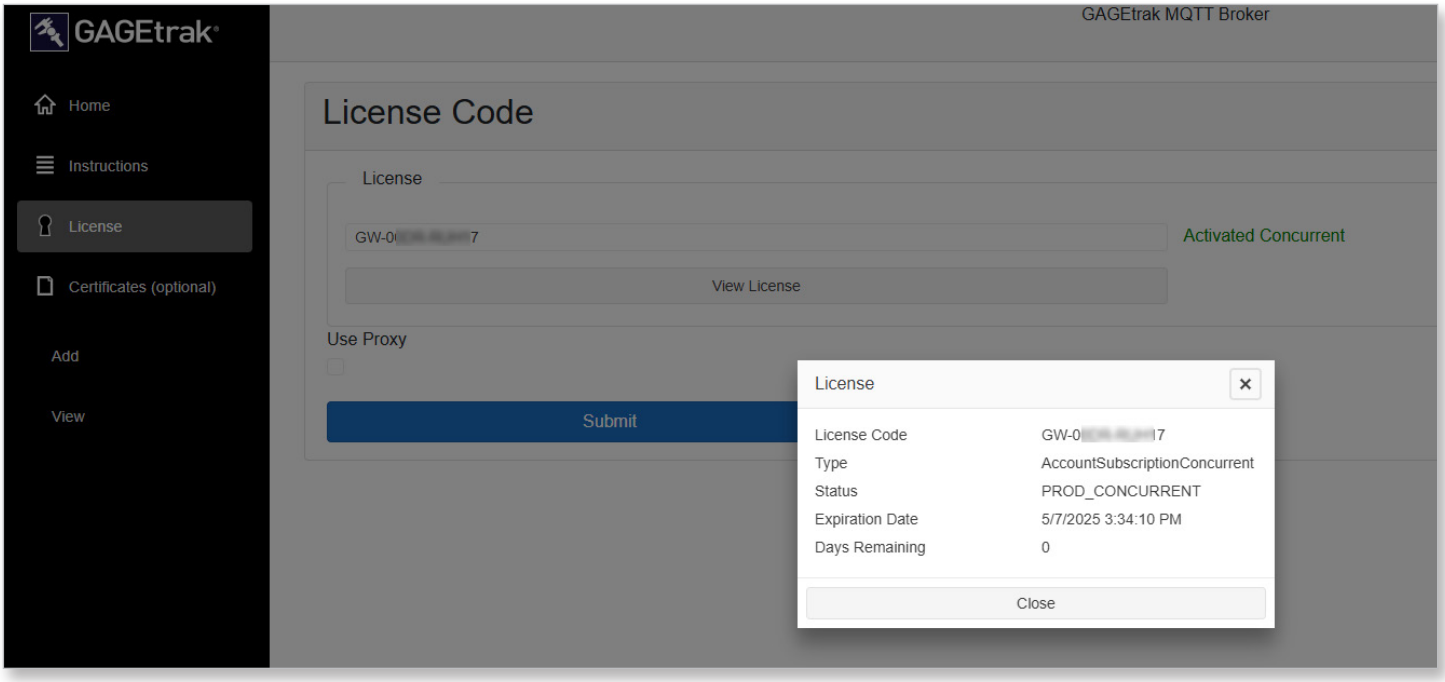
CHAPTER 4: CONFIGURATION

Once the broker installation has been completed, you will need to open a browser to access the broker interface and enter your license code. In your browser's address bar, type ***https://localhost/7214*** where 7214 is one of the ports you selected and noted during installation.

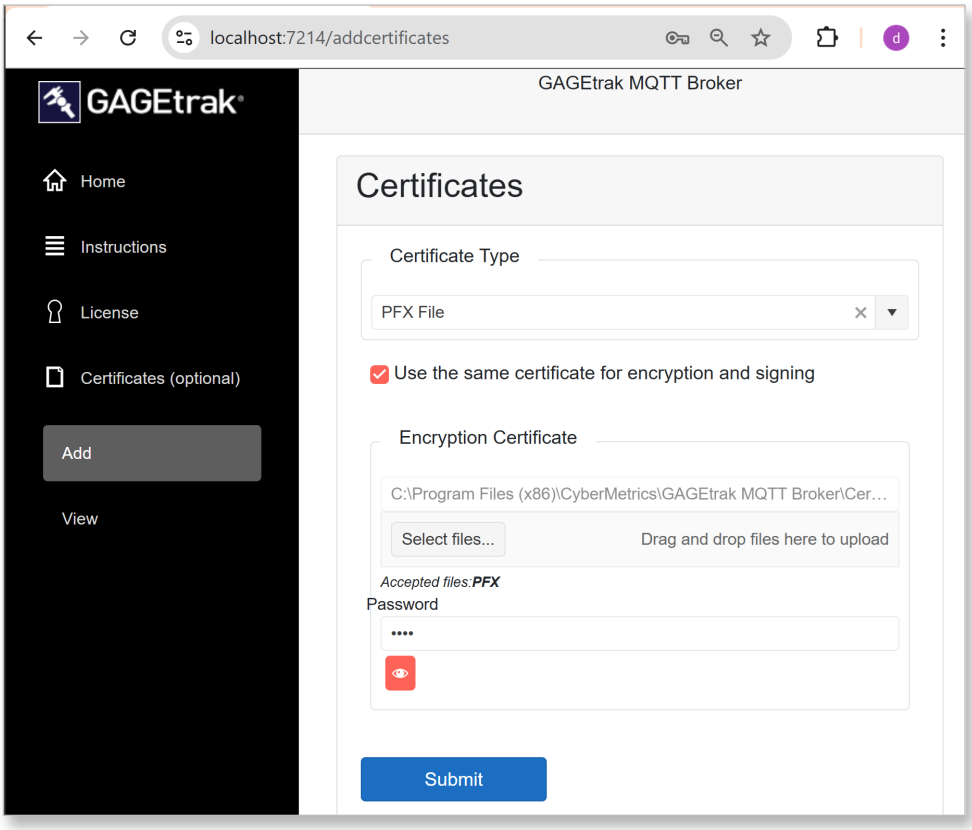
Select **License** on the left, enter your **License Code** and click **Submit**.

Note: If you are unable to reach the broker, be sure you entered the correct port and try restarting the **GAGEtrak MQTT Broker** Windows service.

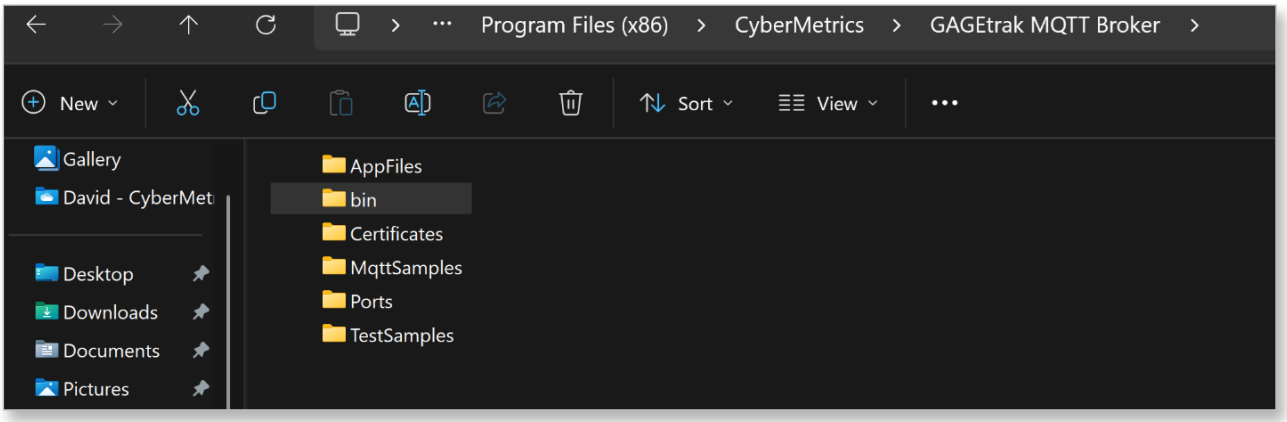
Once activated, the **View License** bar will provide license details.



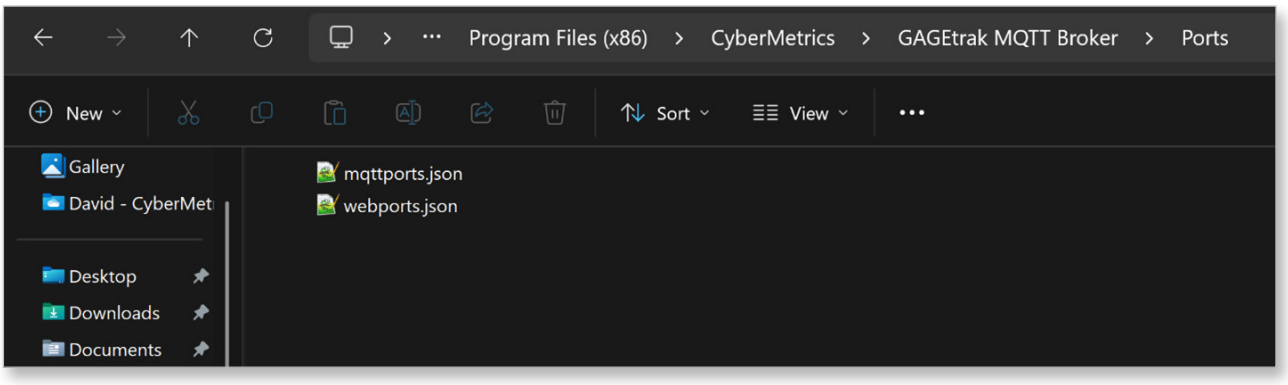
The installation provides self-signed root certificates that can be used. Therefore, setting up your own trusted root certificate is optional. The certificates provided are generated by **Gt.GenerateCert.exe** and can be re-generated, if needed, by running the same executable from the broker install directory, i.e., **C:\Program Files (x86)\CyberMetrics\GAGEtrak MQTT Broker\bin\ Gt.GenerateCert.exe**.



In the GAGEtrak Broker directory, you can review the installed certificates in the **Certificates** directory.

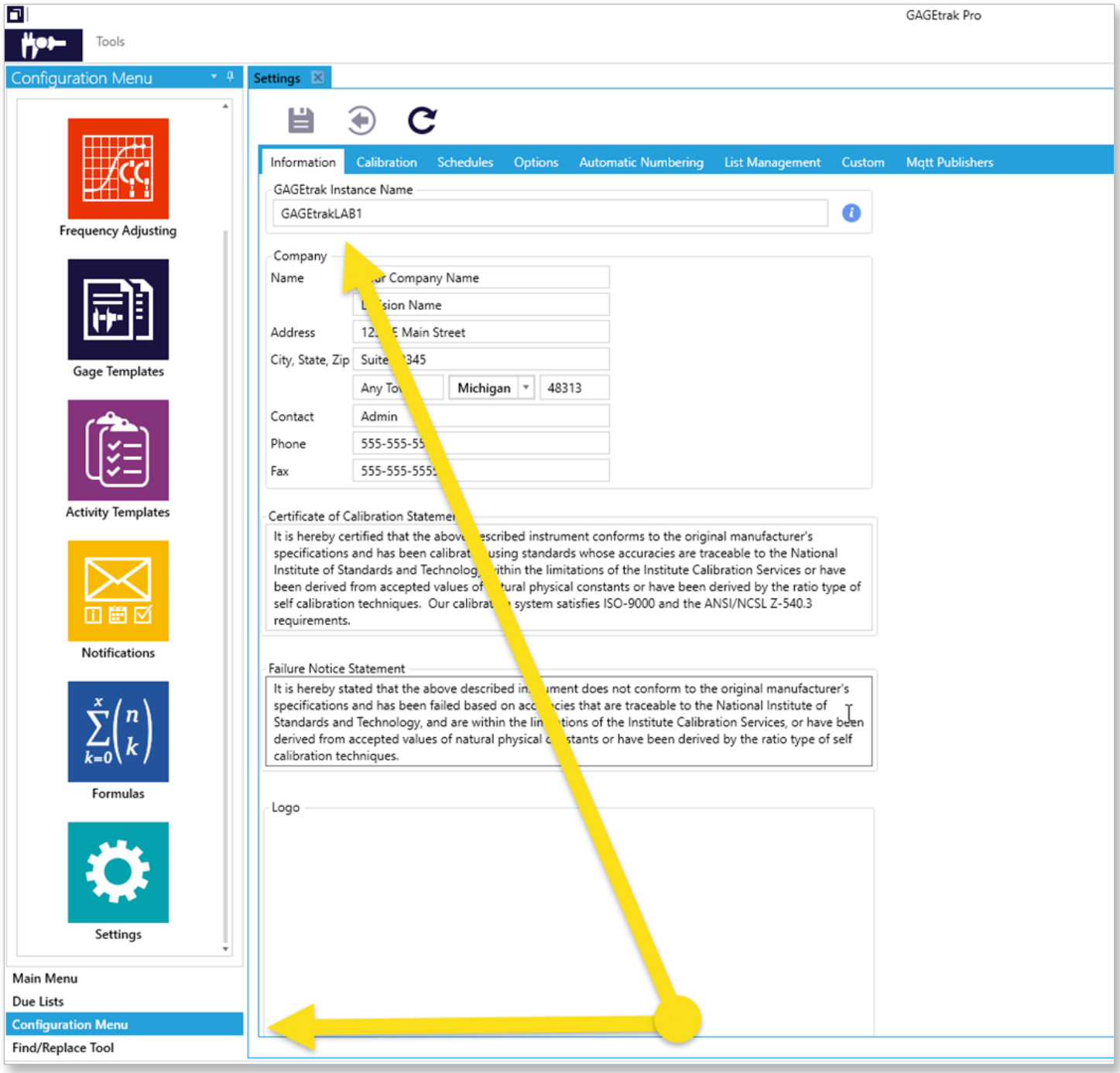


The **Ports** directory contains JSON files that store the MQTT and web port selections made during installation. If you ever forget the ports being used, you can open those files to reference the ports.
Important: You can only change these ports during installation or from these JSON files post install.

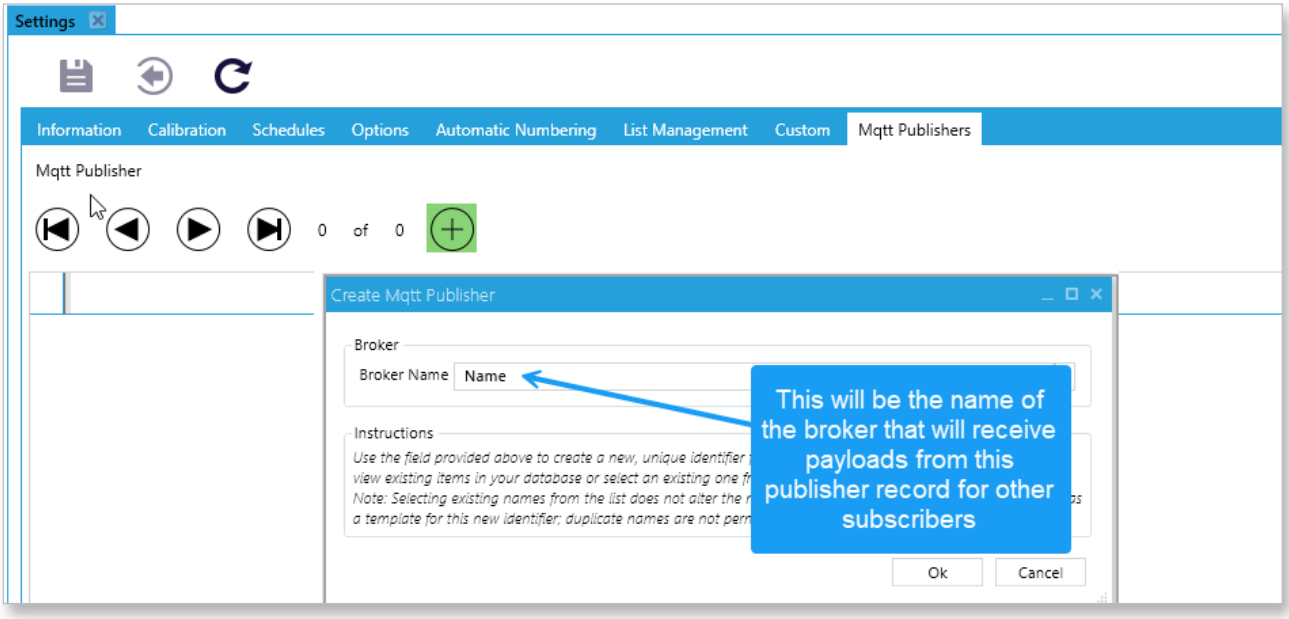


CHAPTER 5: GAGETRAK PUBLISHER CLIENT

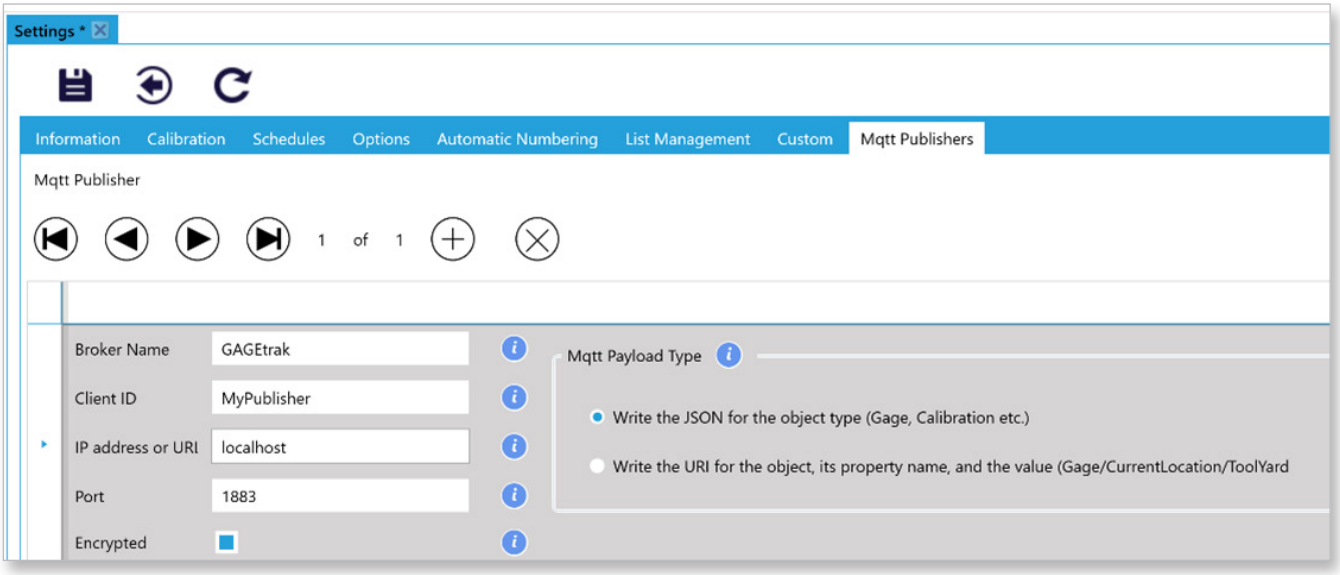
To receive notifications from a GAGEtrak client, launch the GAGEtrak Pro/Lite application and then select **Settings** from the **Configuration Menu**. In the **Information** tab, enter a **GAGEtrak Instance** name; this is a unique name for your GAGEtrak instance, which will be used for MQTT and by other applications. This will appear at the beginning of the topic in your MQTT output, i.e., ***GAGEtrakLAB1/Gage/022E-DCAL***.



Next, select the **MQTT Publishers** tab. Click the ⊕ icon to add a publisher and enter the name of the GAGEtrak Broker in the **Create MQTT Publisher** dialog.



Enter a unique **Client ID**, the **IP** or **URI** for the broker, and the **Port** (we will use the same port selected when installing the GAGEtrak MQTT Broker). The **Broker Name** and the **Client ID** can be any arbitrary name, provided they are unique in your environment, i.e., **Broker Name** ***GAGEtrakBroker1*** and **Client ID** ***LAB1GageChanges***. These are simply identifiers that will make it easier to determine the origin of any published data in your MQTT ecosystem. This is especially important if there are multiple brokers and publishers in your environment.

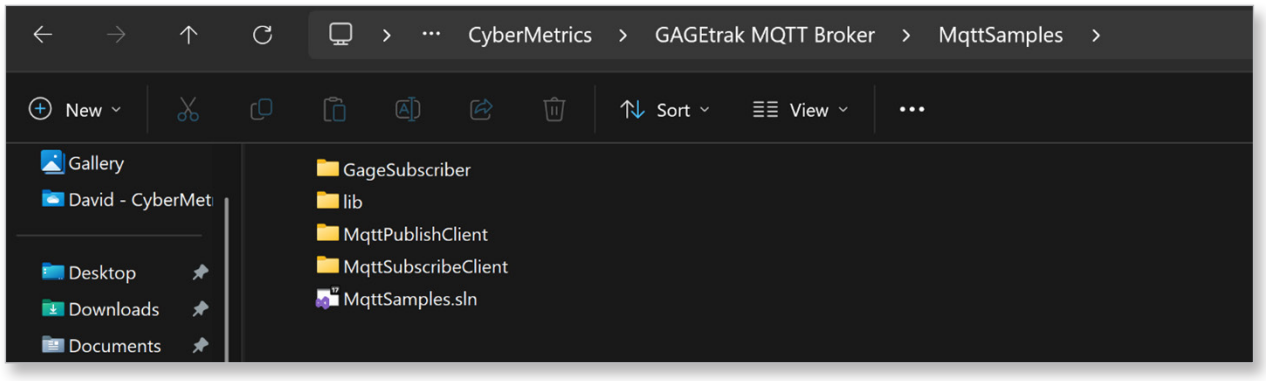


Leave the default **MQTT Payload Type** as JSON. Save the publisher record and you will be ready to receive data from the GAGEtrak client to the broker.

CHAPTER 6: SAMPLE PROJECTS

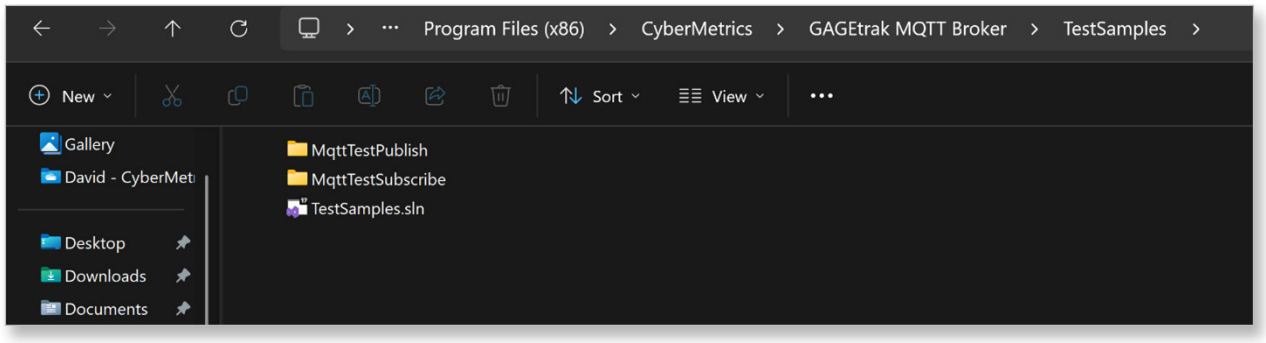
Sample projects have been included with the GAGEtrak MQTT Broker install. These can be found in the installation directory in the **MqttSamples** folder. To create a sample project, you will need a code editor, like Visual Studio or Visual Code, etc. You will also need an installed GAGEtrak client with an **Instance Name** and MQTT Publisher record created in **Settings**.

You can use the starting examples provided to test the broker. The **MqttSamples** folder contains three projects:

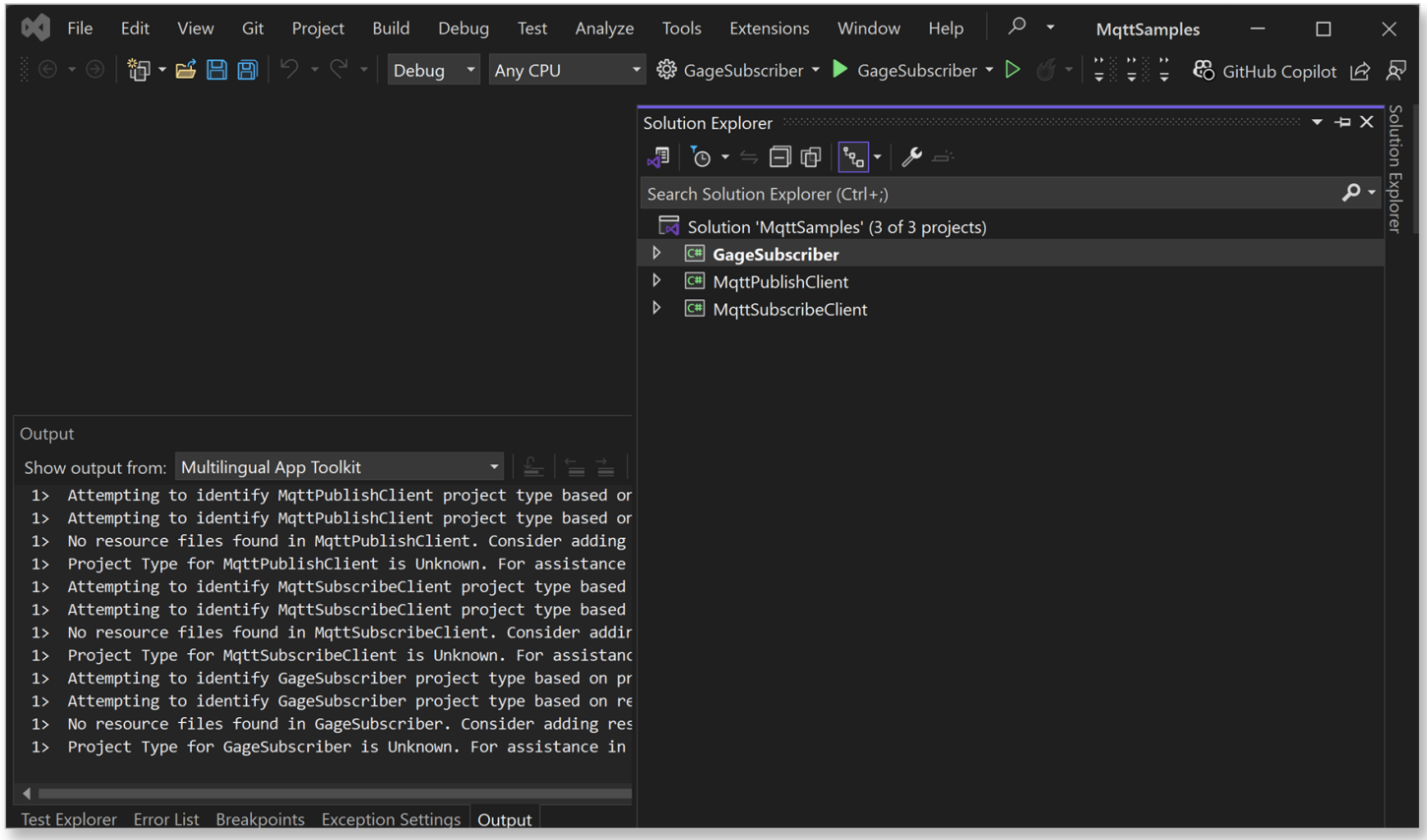


MqttPublishClient and **MqttSubscribeClient** are used to test the broker; they publish and subscribe to a room temperature. We will start the project with the **GageSubscriber**.

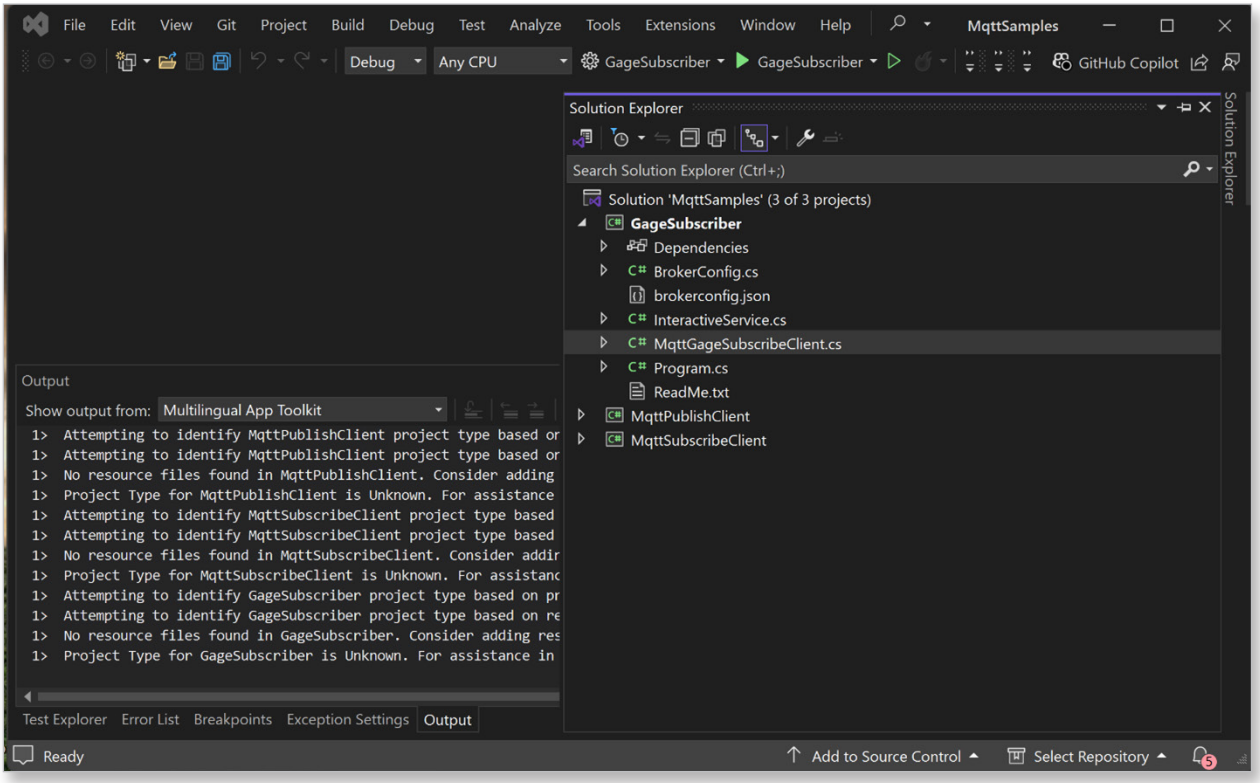
The **TestSamples** directory contains a general publisher and subscriber that is used to do high-speed testing of the broker.



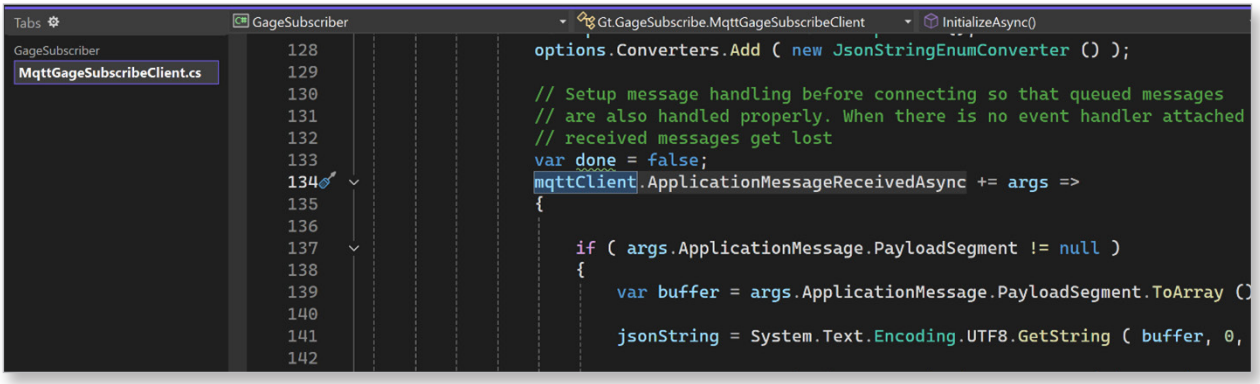
When you are ready to begin testing, go to the installation directory for the broker and in the **MqttSamples** folder, open **MqttSamples.sln** and set **GageSubscriber** as the start-up project.



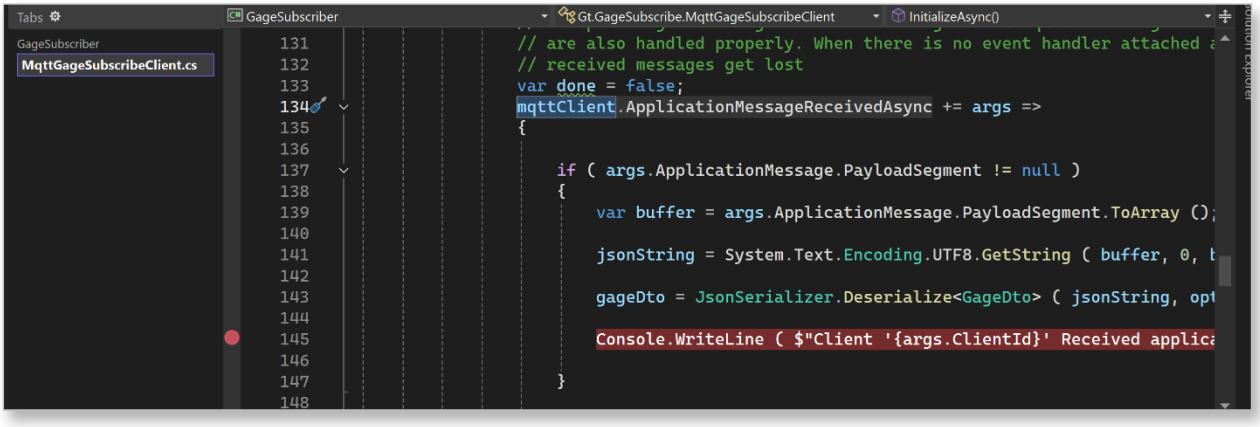
Open the file **MqttGageSubcriberClient.cs**.



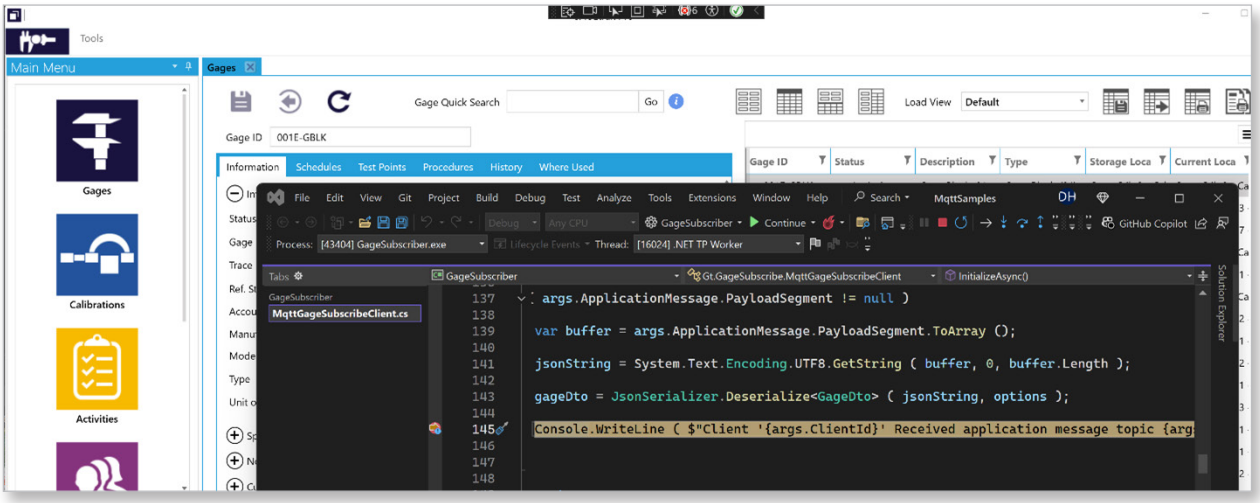
Search for ***mqttClient.ApplicationMessageReceivedAsync*** in the code.



Put a break point after the “if” statement.

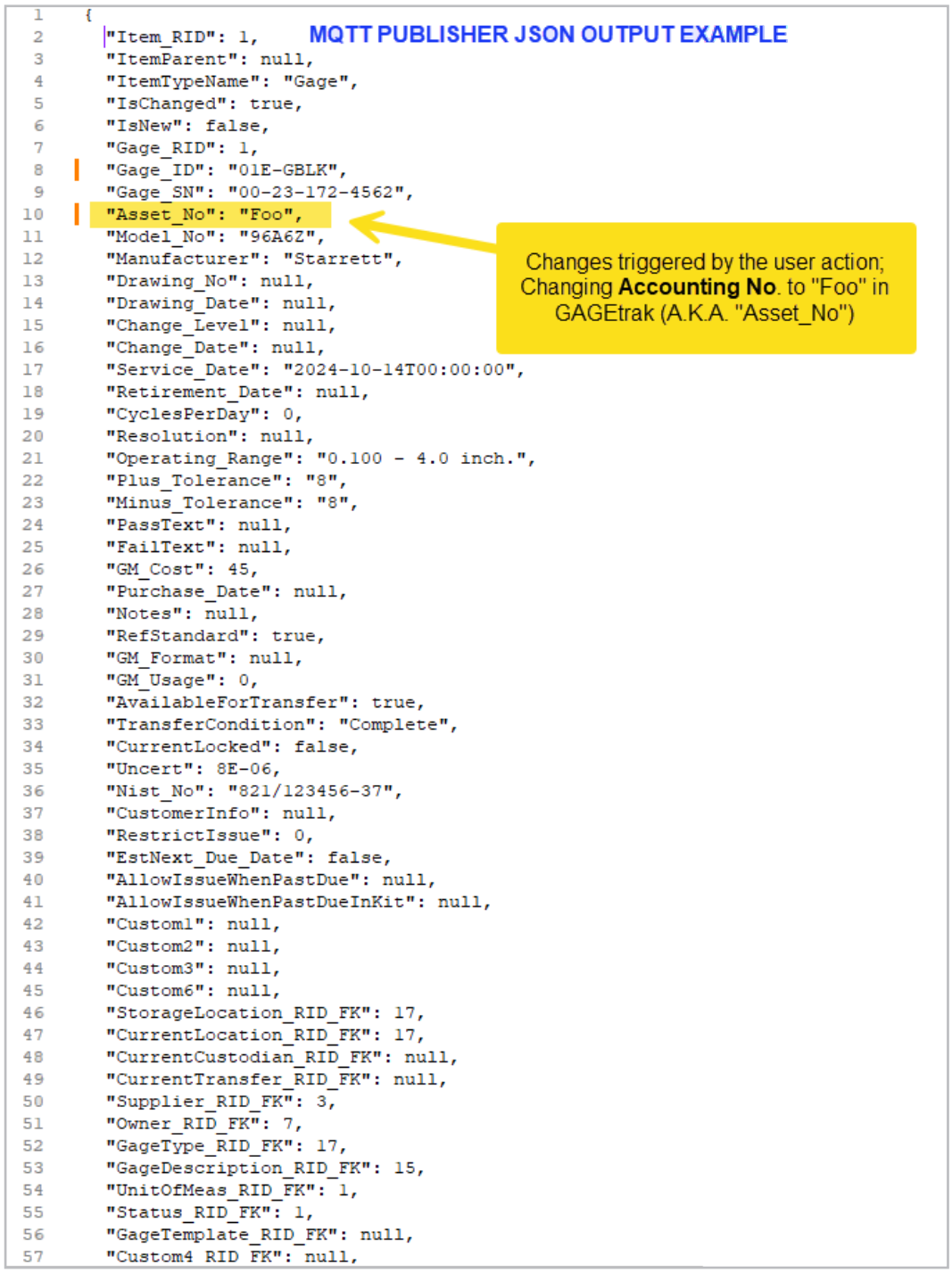


Run the project and start the GAGEtrak Pro/Lite client application. In **Gages**, locate any gage record, change the **Accounting No.** to ***Foo*** and then save. In a few seconds, your break point will be hit.



If you are not seeing the break point or any output, make sure the **GAGEtrak MQTT Broker** Windows Services is running on the broker machine. Verify that your IP address is correct and the MQTT port is not blocked by a firewall on either the broker or GAGEtrak client. Additionally, make certain the GAGEtrak MQTT Broker License Code is activated, otherwise the break point will not be hit. You need a valid license to get subscriptions to the GAGEtrak MQTT Broker.

Below is a sample of data changes published from the Sample Project:




```
{
  "Item_RID": 1,
  "ItemParent": null,
  "ItemTypeName": "Gage",
  "IsChanged": true,
  "IsNew": false,
  "Gage_RID": 1,
  "Gage_ID": "01E-GBLK",
  "Gage_SN": "00-23-172-4562",
  "Asset_No": "Foo",
  "Model_No": "96A6Z",
  "Manufacturer": "Starrett",
  "Drawing_No": null,
  "Drawing_Date": null,
  "Change_Level": null,
  "Change_Date": null,
  "Service_Date": "2024-10-14T00:00:00",
  "Retirement_Date": null,
  "CyclesPerDay": 0,
  "Resolution": null,
  "Operating_Range": "0.100 - 4.0 inch.",
  "Plus_Tolerance": "8",
  "Minus_Tolerance": "8",
  "PassText": null,
  "FailText": null,
  "GM_Cost": 45,
  "Purchase_Date": null,
  "Notes": null,
  "RefStandard": true,
  "GM_Format": null,
  "GM_Usage": 0,
  "AvailableForTransfer": true,
  "TransferCondition": "Complete",
  "CurrentLocked": false,
  "Uncert": 8E-06,
  "Nist_No": "821/123456-37",
  "CustomerInfo": null,
  "RestrictIssue": 0,
  "EstNext_Due_Date": false,
  "AllowIssueWhenPastDue": null,
  "AllowIssueWhenPastDueInKit": null,
  "Custom1": null,
  "Custom2": null,
  "Custom3": null,
  "Custom6": null,
  "StorageLocation_RID_FK": 17,
  "CurrentLocation_RID_FK": 17,
  "CurrentCustodian_RID_FK": null,
  "CurrentTransfer_RID_FK": null,
  "Supplier_RID_FK": 3,
  "Owner_RID_FK": 7,
  "GageType_RID_FK": 17,
  "GageDescription_RID_FK": 15,
  "UnitOfMeas_RID_FK": 1,
  "Status_RID_FK": 1,
  "GageTemplate_RID_FK": null,
  "Custom4_RID_FK": null,
```

CHAPTER 7: CONCLUSION

Modern manufacturing environments require large amounts of real-time data exchange across distributed systems. MQTT (Message Queuing Telemetry Transport) is a lightweight, scalable protocol purpose-built for this challenge. MQTT can specifically support calibration management and broader quality system data transport through decoupled, real-time messaging, unlocking new opportunities in analytics, compliance and digital transformation.

Once you have established the broker and pub/sub configurations, your whole quality organization can begin to reap the benefit.

If you need any assistance working with either the GAGEtrak MQTT Broker or GAGEtrak Publisher component, please reach out to our support team: https://gagetrak.com/support/.

TRAINING

To save time, avoid common mistakes and get the most out of your software, it is imperative to receive proper training and we offer a wide variety of training options to meet every need and budget.

Completion of comprehensive training entitles each participant to a completion certificate in order to provide proof to an auditor that you've received proper training from the software manufacturer.

Go to https://gagetrak.com/training-schedule/ for details about types of training, schedules and pricing.

CONTACT US

Feel free to contact us at any time, for any reason. We're here to help and value your feedback.
1-800-777-7020 https://gagetrak.com/contact-us/