MQTT – Message Queuing Telemetry Transport

Summary Information

Communication Driver Name: MQTT

Implementation DLL: T.ProtocolDriver.MQTT.dll

Interface: TCPIP

Protocol: MQTT (Message Queuing Telemetry Transport) is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol

PC Hardware requirements: Ethernet board

Implemented Methods: Connect, Disconnect, Subscribe, Unsubscribe, and Publish

Channel Configuration

Protocol Options

Not used in this driver

Node Configuration

Station Configuration

Station syntax: <URL>;<Port>;[Username];[Password];[X509Certificate]

Where:

- **<URL>** = A MQTT Broker (Server) name. It must be the same name that is configured in the bind_address parameter in the Broker
- **<Port>** = A MQTT Broker port. It must be the same port configured in the Broker as the listening port. Default value is 1883
- **[Username],** optional = A Username defined on the MQTT Broker. This will be requested if the Broker has the allow_anonymous parameter as false. If this parameter is set as true on the Broker, keep this field empty. Default value is empty
- **[Password],** optional = A Password defined on the MTQQ Broker. This will be requested if the Broker has the allow_anonymous parameter as false. If this parameter is set as true on the Broker, keep this field empty. Default value is empty
- **[X509Certificate],** optional = The Path of the X509 Certificate using TLS v1.0. This must be the complete path of the X509 certificate in the client computer. The certificate must be installed in the computer

> ⚠ **Note**
>
> One of the easiest ways to install the certificate on the client computer is to use the wizard to import the certificates through the "Internet Options". To import the certificate in the "Trusted Root Certification Authorities", you need to import the certificate in the .DER format.

Station Examples

Broker configuration with no Username and Password and no SSL/TLS

Station Configuration

- **URL:** test.mosquitto.org
- **Port:** 1883
- **Username:** <empty>
- **Password:** <empty>
- **X509Certificate:** <empty>

Broker configuration with Username and Password and no SSL/TLS

Station Configuration

- **URL:** MSI
- **Port:** 1883
- **Username:** Guest
- **Password:** 1234
- **X509Certificate:** <empty>

Broker configuration with SSL/TLS and no Username and Password

Station Configuration

- **URL:** MSI
- **Port:** 8883
- **Username:** <empty>
- **Password:** <empty>
- **X509Certificate:** C:\Certificates \mqtt_ca.der

Broker configuration with SSL/TLS and Username and Password

Station Configuration

- **URL:** MSI
- **Port:** 8883
- **Username:** Guest
- **Password:** 1234
- **X509Certificate:** C: \mqtt_ca.de

Point Configuration

Address

The syntax for the MQTT communication points is: <Topic>:<QoS>

Where:

- **<Topic>** is a string, which is used by the broker to filter messages for each connected client. A topic consists of one or more topic levels. Each topic level is separated by a forward slash (topic level separator). Wildcards are not supported.
- **<QoS>** Quality of Service (QoS) is an agreement between the sender and receiver of a message regarding the message's guarantees of delivery. The 3 QoS levels are AtMostOnce, AtLeastOnce, and ExactlyOnce.

E.g.: USA/Houston/Station1/Level:AtMostOnce

Troubleshoot

The status of the driver's execution can be observed through the diagnostic tools, which are:

- Trace window
- Property Watch
- Module Information

The above tools indicate if the operations have succeeded or have failed. A status of 0 (zero) means communication is successful. Negative values indicate internal driver errors, and positive values indicate protocol error codes.