

PRODUCT	DEVICE
RASPI-GO-MQTT	Remote Access VPN Solution + MQTT

RASPI-GO-MQTT

MQTT (MQ Telemetry Transport) protocol provides a lightweight method of carrying out messaging using a publish/subscribe model. This makes it suitable for Internet of Things messaging such as with low power sensors or mobile devices such as phones, embedded computers or microcontrollers.

RASPI-GO-MQTT Broker – Handles many concurrent connected clients and is responsible for receiving all messages, filtering, deciding who is interested and sending the message to subscribed clients.

Broker Addresses for Publishing/Subscribing of data

Local Address = xxx.xxx.xxx.xxx (local IP address & subnet setup for client)

Internet (IOT) Address = NAME.vpnazure.net

How to **publish** a data point to the Broker example

TOPIC - For Niagara USE QoS(1)

- demo/ventilation/sensors/supply_air_temp

How to **subscribe** to published data from the broker.

TOPIC - For Niagara USE QoS(1)

- demo/ventilation/sensors/supply_air_temp

Publish/Subscribe

The MQTT protocol is based on the principle of publishing messages and subscribing to topics, or "pub/sub". Multiple clients connect to a broker and subscribe to topics that they are interested in. Clients also connect to the broker and publish messages to topics. Many clients may subscribe to the same topics and do with the information as they please. The broker and MQTT act as a simple, common interface for everything to connect to. This means that you if you have clients that dump subscribed messages to a database, then it becomes very simple to add new sensors or other data input to a database.

Topics/Subscriptions

Messages in MQTT are published on topics. There is no need to configure a topic, publishing on it is enough. Topics are treated as a hierarchy, using a slash (/) as a separator. This allows sensible arrangement of common themes to be created, much in the same way as a filesystem. For example, multiple computers may all publish their hard drive temperature information on the following topic, with their own computer and hard drive name being replaced as appropriate:

sensors/COMPUTER_NAME/temperature/HARDDRIVE_NAME

Quality of Service

MQTT defines three levels of Quality of Service (QoS). The QoS defines how hard the broker/client will try to ensure that a message is received. Messages may be sent at any QoS level, and clients may attempt to subscribe to topics at any QoS level. This means that the client chooses the maximum QoS it will receive. For example, if a message is published at QoS 2 and a client is subscribed with QoS 0, the message will be delivered to that client with QoS 0. If a second client is also subscribed to the same topic, but with QoS 2, then it will receive the same message but with QoS 2. For a second example, if a client is subscribed with QoS 2 and a message is published on QoS 0, the client will receive it on QoS 0.

Higher levels of QoS are more reliable, but involve higher latency and have higher bandwidth requirements.

- 0: The broker/client will deliver the message once, with no confirmation.
- 1: The broker/client will deliver the message at least once, with confirmation required.
- 2: The broker/client will deliver the message exactly once by using a four-step handshake.