

# 03 Edge 2.1

## MQTT API Reference Guide

Edition 1.0

© 2021 Delta Controls All rights reserved.

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language (natural or computer), in any form or by any means, without the prior written permission of Delta Controls.

Limited permission is granted to reproduce documents released in Adobe® Portable Document Format (PDF) electronic format in paper format. Documents released in PDF electronic format may be printed by end-users for their own use using a printer such as an inkjet or laser device. Authorized distributors of Delta Controls products (Delta Partners) may print PDF documents for their own internal use or for use by their customers. Authorized Delta Partners may engage a printing or copying company to produce copies of released PDF documents with the prior written permission of Delta Controls.

Information in this document is subject to change without notice and does not represent a commitment to past versions of this document on the part of Delta Controls. Delta Controls may make improvements and/or changes to this document at any time.

Document edition: 1.0

# Contents

- Introduction** ..... 1
  - About the O3 Edge ..... 1
  - About MQTT ..... 1
  - MQTT and the O3 Edge ..... 3
    - Topic Format ..... 3
    - Setting up EnOcean Devices ..... 4
    - Message Format ..... 4
- Getting Started** ..... 5
  - Enable the Internal MQTT Broker ..... 5
  - Mobile MQTT Client Example ..... 5
    - Connect to the Broker ..... 5
    - Subscribe to a Topic ..... 7
  - Desktop MQTT Client Example ..... 8
    - Connect to the Broker ..... 8
    - Publish to a Topic ..... 9
- MQTT Topics** ..... 10
  - Subscribe Topics ..... 10
    - Object Topics ..... 10
    - Metrics Topics ..... 14
  - Publish Topics ..... 16
    - Object Topics ..... 16
    - Metrics Topics ..... 18
    - Firmware and Reboot Topics ..... 18
  - Setting Topics ..... 19
- BACnet Secure Connect** ..... 21
  - Setting up Secure Connect using MQTT ..... 21
  - Set the Auto Config URL ..... 21
  - Enable Secure Connect connection ..... 21
  - Approve device in enteliWEB ..... 22
  - Check connection status ..... 22

Disable Secure Connect Connection .....	22
<b>Using MQTT with EnOcean Devices .....</b>	<b>23</b>
Registering Objects .....	23
Reading Objects .....	23
Writing Objects .....	23
<b>Revision History .....</b>	<b>24</b>

# Introduction

This guide describes the MQTT topics associated with the O3 Edge series edge controller. These topics are used to publish and read data from the device and send commands to control outputs.

The information in this guide applies to the following models:

- O3-EDGE-02
- O3-EDGE-02-E8
- O3-EDGE-02-E9

## About the O3 Edge

The O3 Edge ("O3") is a ceiling-mounted multisensor device that integrates temperature, humidity, motion, sound, and light sensing. Using sensor fusion technology and machine learning algorithms, the O3 delivers fast, accurate feedback on the monitored space.

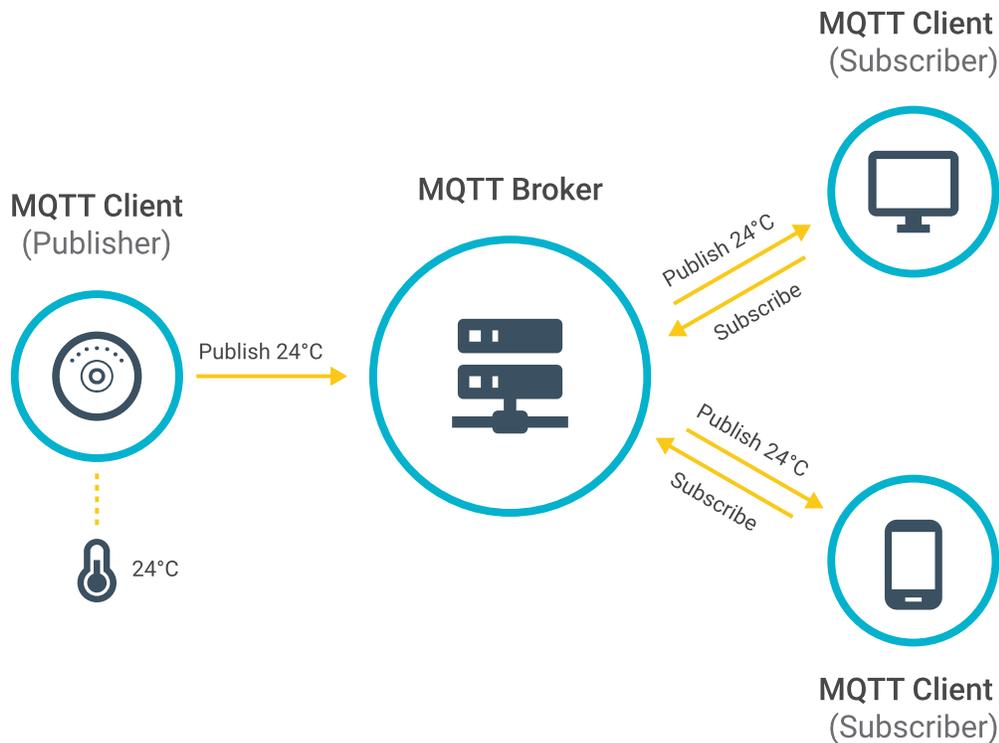
The O3 supports BACnet, MQTT, EnOcean, and Bluetooth® Low Energy (BLE) protocols, allowing it to integrate with almost any system. It also comes with two universal I/O points for controlling or accepting inputs from nearby devices.

For more information about the O3, go to <https://deltacontrols.com/o3gettingstarted/>.

## About MQTT

MQTT is a lightweight publish/subscribe messaging protocol built on TCP/IP. Originally designed for monitoring oil pipelines (as *MQ Telemetry Transport*), MQTT is now the standard messaging and data exchange protocol for the Internet of Things.

An MQTT **client** is any device that runs an MQTT library and connects to an MQTT **broker** (server) over a network. Unlike in the traditional client-server model where clients communicate directly with an endpoint, in the publish/subscribe ("pub/sub") model, clients communicate through a central **broker**. The client that sends a message (the **publisher**) and the client that receives that message (the **subscriber**) never interact. The broker filters incoming messages from the publisher and distributes them to the subscriber(s).



Messages are filtered using topics. A **topic** is a UTF-8 string that the broker uses to decide which subscriber receives which message. The topic consists of one or more topic levels, with each topic level separated by a forward slash (topic level separator). A topic must be at least one character long and topic names are case-sensitive.

For example, in the topic `events/object/occupantTemperature`, the first level is "events", the second level is "object", and the third level is "occupantTemperature".

When a client subscribes to a topic, it can subscribe to the exact topic of a published message or it can use wildcards to subscribe to multiple topics at once. There are two types of **wildcards**: single-level (+) and multi-level (#). The single-level wildcard replaces one topic level, while the multi-level wildcard replaces multiple topic levels. The multi-level wildcard must always be the last character in the topic string and must be preceded by a forward slash.

For example: `events/object/#`

 Wildcards cannot be used when publishing to a topic.

Senders and receivers set a **Quality of Service (QoS)** level that defines how much effort will be expended to ensure that the message is delivered.

There are three QoS levels in MQTT:

Level 0	The message is delivered <i>at most once</i> , according to the best efforts of the underlying TCP/IP network. There is no guarantee of delivery. You can think of this approach as send and forget.
Level 1	The message is delivered <i>at least once</i> , with multiple retries until the message is acknowledged as received. You can think of this approach as acknowledged delivery.
Level 2	The message is delivered <i>exactly once</i> , with the sender and receiver doing a handshake to ensure that the message is received once by the intended recipient. You can think of this approach as guaranteed delivery.

Both the subscriber and publisher can set a QoS level. If the levels don't match, the lower of the two is used. For example, if a client subscribes to a message at a QoS of 1 but the message was published at a QoS of 0, then the client will receive the message at a QoS of 0.

The content of a message is referred to as the **payload**. For example, the payload of a subscribe message will contain a list of topic names to which the client wants to subscribe, as well as the QoS level at which the client wants to receive the messages.

For more information about MQTT, go to <https://mqtt.org>.

## MQTT and the O3 Edge

 The O3 currently supports the [MQTT v3.1.1 specification](#).

The MQTT topics associated with the O3 Edge can be used to subscribe to data from the O3's internal [Eclipse Mosquitto™](#) broker and send commands to change the O3's setpoints and outputs.

### Topic Format

The O3's sensor and device properties are represented by BACnet objects, which are mapped to MQTT topics through an internal BACnet-to-MQTT bridge. The MQTT topics are divided into "events" topics and "commands" topics, indicating the direction of the message. "Events" topics (`events/object/SUBTOPIC`) are used to report BACnet object values (from sensors, outputs, etc.), while "commands" topics (`commands/object/SUBTOPIC`) are used to set BACnet object values.

To learn more about the BACnet objects associated with the O3 Edge, refer to the *O3 Edge BACnet Application Guide* on [support.o3hub.com](https://support.o3hub.com).

## Setting up EnOcean Devices

The O3 supports up to 32 EnOcean wireless devices. To access EnOcean objects, you must first create a device SV object for each EnOcean device you want to configure. After which you can set EnOcean as your gateway mapping for each device, and then associate a mapping file with each SV object. For more information on setting up EnOcean devices using enteliWEB, refer to the *O3 Edge EnOcean Application Guide* on <https://deltacontrols.com/o3gettingstarted/>.

After you've configured your objects in enteliWEB, you can then use MQTT to read and write EnOcean objects. For more information, refer to [Using MQTT With EnOcean Devices](#).

## Message Format

The O3 supports key-value content in JSON (JavaScript Object Notation) format as follows:

```
{"key1": "value1"}
```

The key is always a string, while the value can be either a string, number, boolean, object, array, or null. Spaces before or after the colon are optional, as JSON ignores whitespace between elements.

For example, the `events/object/occupantTemperature` topic looks like this:

```
{"Present_Value": 24.51, "Units": "°C", "updated": "31-08-2021  
23:09:26.21", "status": 0}
```

# Getting Started

This chapter describes how to enable and connect to the O3's internal MQTT broker and subscribe/publish to topics using a third-party MQTT client application.

## Enable the Internal MQTT Broker

The internal MQTT broker is disabled by default. To use a mobile or desktop MQTT client, you will need to enable the internal MQTT broker. To enable the internal MQTT broker, you can use the O3 Setup mobile app or enteliWEB.

To enable the internal MQTT broker using the O3 Setup mobile app:

1. After connecting to the O3, go to **Settings > MQTT (Internal)** and select .
2. Select **Enable MQTT broker**, then select **Save**.
3. Select **Apply settings to device**.

To enable the internal MQTT broker using enteliWEB:

- Command object **BV48** (Internal MQTT Broker Enable) to manual mode and set the object value to **On**, then click **Save**.

## Mobile MQTT Client Example

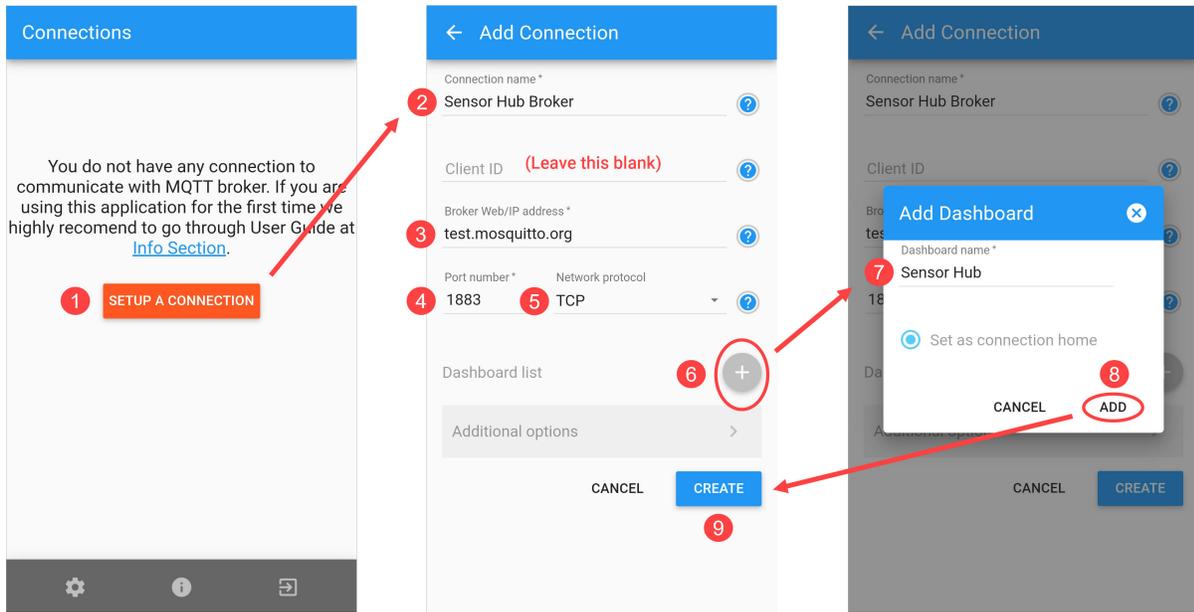
For this example, we will use IoT MQTT Panel, a free mobile app available on Google Play.

### Connect to the Broker

Open the IoT MQTT Panel app and tap **Setup a Connection**. Enter the name, address, and port of the MQTT broker that you are connecting to, add a dashboard, and then tap **Create**.

- To connect to the O3's internal MQTT broker, use the device's IP address.
- The default MQTT port over TCP is **1883**.
- To publish to the O3's internal MQTT broker, you will need to enter a user name and password. If you are publishing to the O3's internal MQTT broker for the first time, use the following login credentials: username = **user**, password = **admin**. The username and password are case-sensitive. You can enter this information under **Additional options**. After logging in, change the default password in [setting/mqttPassword](#).

# Getting Started

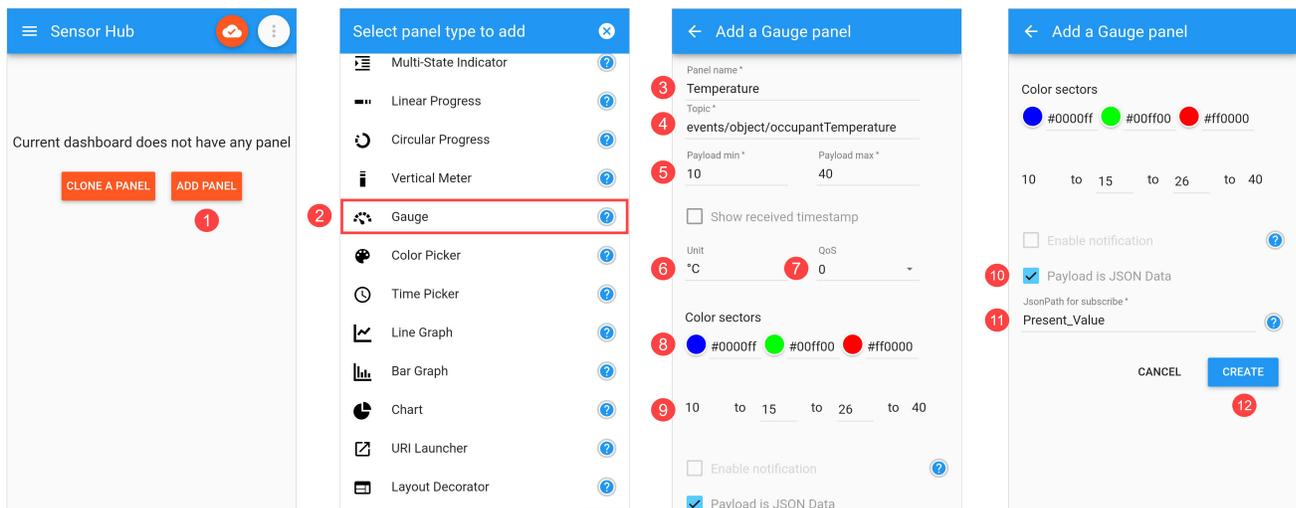


## Subscribe to a Topic

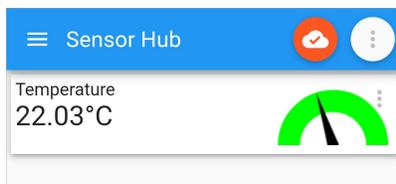
On the Connections page, you should see the connection that you have created. Tap the cloud icon to connect to the broker. The cloud icon changes color and shows a check mark when a connection to the broker is established.

Next, the dashboard page opens and you are prompted to create a panel. Tap **Add Panel**, select a panel type, enter the details for the panel, and then tap **Create**.

- Subscribe topics on the O3 follow the format `events/object/SUBTOPIC`. For example, `events/object/occupantTemperature`.
- The JSON key you want to read is `Present_Value`.



The created panel looks like this:



## Desktop MQTT Client Example

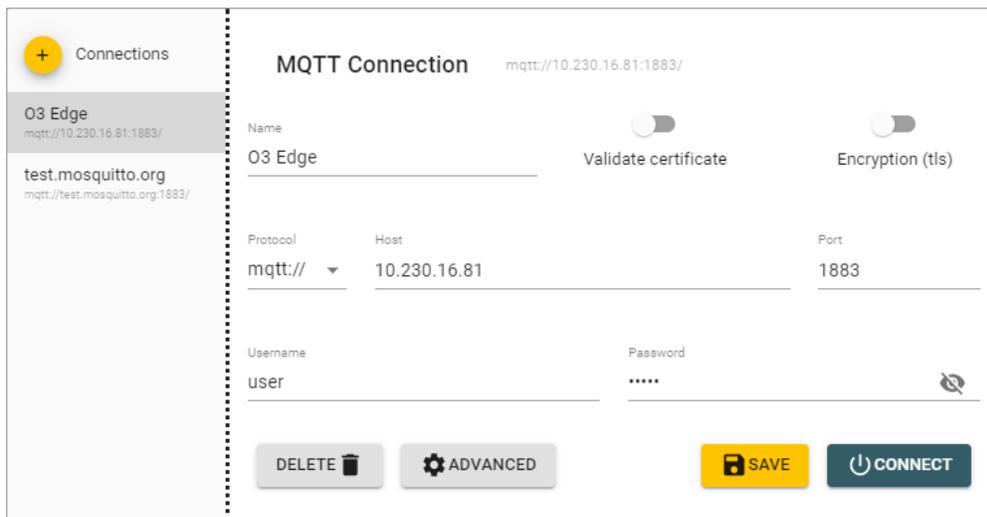
For this example, we will use MQTT Explorer, a desktop MQTT client that lets you visualize, publish, subscribe, and plot topics. You can download it from <http://mqtt-explorer.com>.

### Connect to the Broker

Open MQTT Explorer and click the **Connections** button to add a new connection. Enter the name, address, and port of the MQTT broker.

- To connect to the O3's internal MQTT broker, use the device's IP address.
- The default MQTT port over TCP is **1883**.

In this example, we will be publishing to a topic, so we will also enter a user name and password. Click **Save** to save the settings, then click **Connect**.



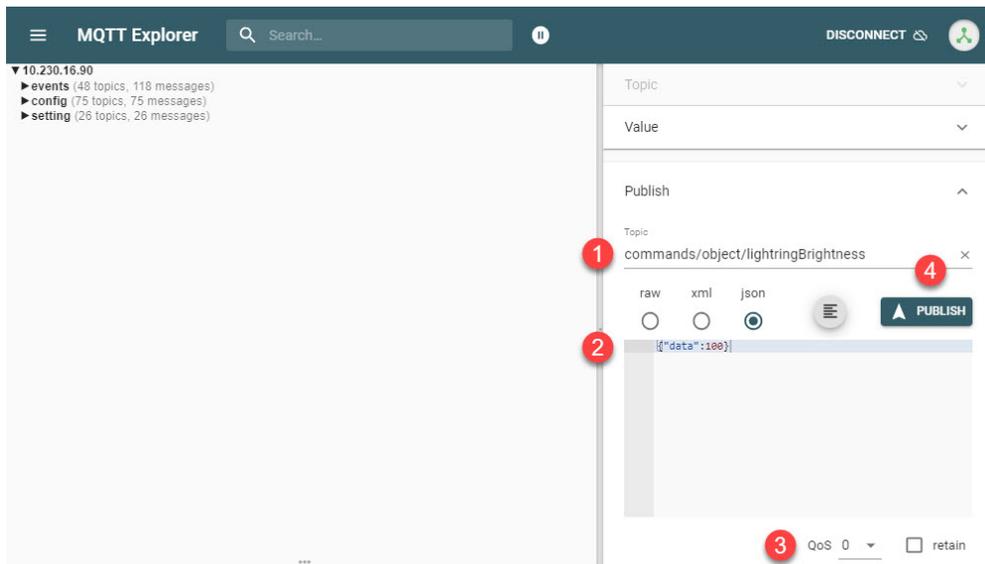
The screenshot shows the MQTT Explorer interface. On the left, there is a sidebar with a 'Connections' button (a yellow circle with a plus sign) and two listed connections: 'O3 Edge' (mqtt://10.230.16.81:1883/) and 'test.mosquitto.org' (mqtt://test.mosquitto.org:1883/). The main area is titled 'MQTT Connection' and shows the configuration for the selected 'O3 Edge' connection. The address bar shows 'mqtt://10.230.16.81:1883/'. Below this, there are fields for 'Name' (O3 Edge), 'Validate certificate' (toggle off), and 'Encryption (tls)' (toggle off). There are also fields for 'Protocol' (mqtt://), 'Host' (10.230.16.81), and 'Port' (1883). At the bottom, there are fields for 'Username' (user) and 'Password' (masked with dots). At the very bottom, there are four buttons: 'DELETE' (trash icon), 'ADVANCED' (gear icon), 'SAVE' (yellow button with a save icon), and 'CONNECT' (dark blue button with a power icon).

 If you are publishing to the O3's internal MQTT broker for the first time, use the following login credentials: username = **user**, password = **admin**. The username and password are case-sensitive. After logging in, change the default password in [setting/mqttPassword](#).

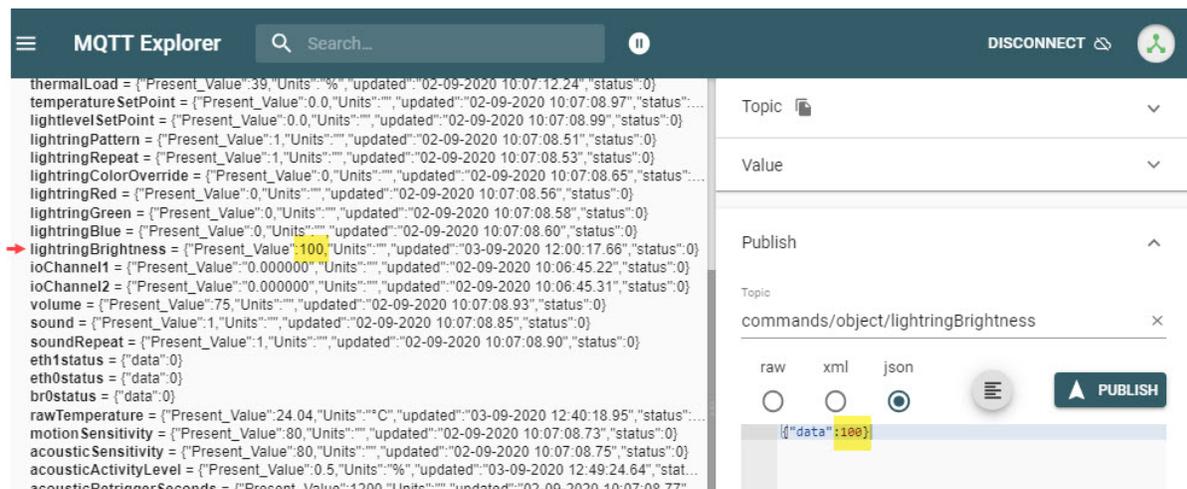
## Publish to a Topic

The main window is divided into two panes. On the left is a tree view of topics. On the right are panels for Topic, Value, Publish, and Stats. Expand the Publish panel and enter the topic that you want to publish to. Select **json** and enter the key-value pair that you want to modify. The value should be the new value that you want to publish. Select a **QoS** level, then click **Publish**.

- Publish topics on the O3 follow the format `commands/object/SUBTOPIC`. For example, `commands/object/lightringBrightness`.
- The JSON key you want to write to is `data`.



After you click **Publish**, you can look up the corresponding subtopic under `events/objects` and verify that the `Present_Value` has changed.



# MQTT Topics

This chapter lists the MQTT topics associated with the O3 Edge. You can subscribe to "events" topics and publish to "commands" topics.

## Subscribe Topics

### Object Topics

You can subscribe to read-only object topics that contain sensor data and device configurations. Subscribe using the format `events/object/SUBTOPIC` and read the `Present_Value` JSON key. To subscribe to all object topics, use `events/object/#`.

TABLE 1: SENSOR DATA OBJECT SUBSCRIBE TOPICS

Topic	Description
<code>events/object/occupantTemperature</code>	Temperature at 1 m (3 ft) above the floor. This is a composite value derived from the O3's internal temperature sensors and the IR temperature sensor. Range: 0°C to 59°C (32°F to 138°F).
<code>events/object/irTemperature</code>	Average temperature of surfaces in the O3's field of view. Range: 0°C to 59°C (32°F to 138°F).
<code>events/object/internalTemperature</code>	Temperature at ceiling height. Range: 0°C to 59°C (32°F to 138°F).
<code>events/object/rawTemperature</code>	Uncalibrated occupant temperature. Range: 0°C to 59°C (32°F to 138°F).
<code>events/object/occupantHumidity</code>	Humidity at 1 m (3 ft) above floor. This is calculated from the occupant temperature and internal humidity using psychrometrics. Range: 0% to 100%.
<code>events/object/internalHumidity</code>	Humidity at ceiling height. Range: 0% to 100%.

TABLE 1: SENSOR DATA OBJECT SUBSCRIBE TOPICS

Topic	Description
events/object/combinedOccupancy	Combined (motion + sound) occupancy signal. Active state when motion and sound is detected. Range: 0 (not occupied), 1 (occupied).
events/object/motion	Motion occupancy signal. Active state when motion is detected. Range: 0 (not occupied), 1 (occupied).
events/object/acousticActivityLevel	Audio level after certain frequencies are filtered out. Range: 0 to 65535.
events/object/acousticOccupancy	Acoustic occupancy signal. Active state when audio level is above acoustic occupancy threshold. Range: 0 (not occupied), 1 (occupied).
events/object/motionSensitivity	Controls the sensitivity of the PIR sensor to changes in movement levels within the detection area. Range: 0% to 100%. 100% = maximum sensitivity.
events/object/acousticSensitivity	Controls the sensitivity of the acoustic occupancy sensor to changes in audio levels within the detection area. Range: 0% to 100%. 100% = maximum sensitivity.
events/object/acousticRetriggerSeconds	The amount of time that activity sounds can cause the O3 to remain in the occupied state after motion is detected. Default value is 1200 seconds (20 minutes). Measured from most recent motion detection event.
events/object/occupancyInactivitySeconds	The amount of time it takes the O3 to return to the unoccupied state when no motion and no audio activity is detected. Default value is 30 seconds (5 minutes).

TABLE 1: SENSOR DATA OBJECT SUBSCRIBE TOPICS

Topic	Description
events/object/acousticBGUpdateSeconds	Update period for the baseline microphone levels to adjust to environmental changes when no occupants are present. Default value is 30 seconds.
events/object/lightLevel	Brightness of ambient light (lux or foot-candle). Range: 0 to 65535 (lux) or 0 to 6088 (foot-candle).
events/object/colorTemperature	Color temperature of ambient light (K). Range: 0 to 65535.
events/object/colorRed	Red component of ambient light. Range: 0 to 65535.
events/object/colorGreen	Green component of ambient light. Range: 0 to 65535.
events/object/colorBlue	Blue component of ambient light. Range: 0 to 65535.
events/object/soundLevel	Level of ambient noise (dB SPL). Unfiltered audio level across the entire spectrum. Range: 0 to 120 dB.

TABLE 2: DEVICE CONFIGURATION OBJECT SUBSCRIBE TOPICS

Topic	Description
events/object/temperatureSetPoint	User-entered temperature. Measured by user at occupant height. Offset is calculated by the O3.
events/object/lightlevelSetPoint	User-entered light level. Records the light level read by the O3 (AI12) when the lighting in the space is set to the desired brightness. This setpoint can be retrieved later by the control system to set the feedback loop, etc.
events/object/lightringPattern	Pre-defined light ring pattern (1-13). Default value is 1 (Off).
events/object/lightringRepeat	Number of times light ring pattern repeats. Default value is 1.
events/object/lightringRed	Red component of light ring RGB value. Range: 0% to 100%. Default value is 0% (Off).
events/object/lightringGreen	Green component of light ring RGB value. Range: 0% to 100%. Default value is 0% (Off).
events/object/lightringBlue	Blue component of light ring RGB value. Range: 0% to 100%. Default value is 0% (Off).
events/object/lightringColorOverride	Custom light ring color defined by lightringRed, lightringBlue, and lightringGreen subtopics. Range: 0 (Off), 1 (On). When set to 1 (On), it overrides lightringPattern.
events/object/lightringBrightness	Overall brightness of light ring. Range: 0% to 100%. Default value is 50%.
events/object/soundinfo	Lists the names and index numbers of all the sound files and their total size. There are 25 default sounds and you can create up to 25 custom sounds. For more details about creating custom sounds, see the <i>O3 Edge BACnet Application Guide</i> on <a href="http://support.o3hub.com">support.o3hub.com</a> . This topic is updated automatically whenever a custom sound file is added, changed, or removed.

TABLE 2: DEVICE CONFIGURATION OBJECT SUBSCRIBE TOPICS

Topic	Description
events/object/volume	Speaker volume. Range: 0% to 100%. Default value is 75%.
events/object/soundRepeat	Number of times a sound is played. Default value is 1.
events/object/startupSoundEnable	Startup sound. Range: 0 (Off), 1 (On). Default value is 0 (Off).
events/object/bleMac	Displays the MAC address of Bluetooth LE beacon.
events/object/ioChannel1	Displays the xP1 universal point value.
events/object/ioChannel2	Displays the xP2 universal point value.

## Metrics Topics

Edge analytics collected on the O3 are known as *metrics*. To subscribe to specific read-only metrics topics from the O3, use the format `events/metrics/SUBTOPIC` and read the `Present_Value` JSON key. To subscribe to all metrics topics, use `events/metrics/#`.

Metrics are computed using a rolling database of archived sensor information, providing immediate insight into room conditions without the need for post-processing.

The O3 archives temperature, humidity, light level, sound level, and occupancy data every time there is a change of value. This archived data is then analyzed hourly to compute the following metrics:

- min/mean/max occupant temperature
- min/mean/max occupant humidity
- min/mean/max light level
- min/mean/max sound level
- occupancy utilization

Once the metrics have been computed, the raw sensor data is purged. The metrics are kept for 30 days. To download this history, publish `{"data": 1}` to `commands/metrics/history`.

TABLE 3: METRICS SUBSCRIBE TOPICS

Topic	Description
events/metrics/occupantTemperature/mean	Average occupant temperature during previous hour.
events/metrics/occupantTemperature/min	Lowest occupant temperature during previous hour.
events/metrics/occupantTemperature/max	Highest occupant temperature during previous hour.
events/metrics/occupantHumidity/mean	Average occupant humidity during previous hour.
events/metrics/occupantHumidity/min	Lowest occupant humidity during previous hour.
events/metrics/occupantHumidity/max	Highest occupant humidity during previous hour.
events/metrics/lightLevel/mean	Average light level during previous hour.
events/metrics/lightLevel/min	Lowest light level during previous hour.
events/metrics/lightLevel/max	Highest light level during previous hour.
events/metrics/soundLevel/mean	Average sound level during previous hour.
events/metrics/soundLevel/min	Lowest sound level during previous hour.
events/metrics/soundLevel/max	Highest sound level during previous hour.
events/metrics/combinedOccupancy/utilization	Percentage of time the space was occupied during the previous hour.

# Publish Topics

## Object Topics

Publish to an object topic using the format `commands/object/SUBTOPIC` and write to the `data` JSON key.

For example, to change the brightness of the O3's LED light ring from 50% to 100%, publish `{"data": 100}` to `commands/object/lightringBrightness`. The corresponding subscribe topic `events/object/lightringBrightness` will then show `{"Present_Value": 100}`.

TABLE 4: OBJECT PUBLISH TOPICS

Topic	Description
<code>commands/object/lightringPattern</code>	Plays pre-defined light ring pattern (1-13). Default value is 1 (Off).
<code>commands/object/lightringRepeat</code>	Sets number of times light ring pattern repeats. Default value is 1.
<code>commands/object/lightringRed</code>	Sets red component of light ring RGB value. Range: 0 to 100. Default value is 0 (Off).
<code>commands/object/lightringGreen</code>	Sets green component of light ring RGB value. Range: 0 to 100. Default value is 0 (Off).
<code>commands/object/lightringBlue</code>	Sets blue component of light ring RGB value. Range: 0 to 100. Default value is 0 (Off).
<code>commands/object/lightringColorOverride</code>	Activates custom light ring color defined by <code>lightringRed</code> , <code>lightringBlue</code> , and <code>lightringGreen</code> subtopics. Set to 1 (On) or 0 (Off). When set to 1 (On), it overrides <code>commands/object/lightringPattern</code> .
<code>commands/object/lightringBrightness</code>	Sets overall brightness of light ring. Range: 0 to 100. Default value is 50.

TABLE 4: OBJECT PUBLISH TOPICS

Topic	Description
commands/object/indicateStatus	Displays power-on self-test (POST) status. Set to 1 (On) or 0 (Off). POST status is indicated by the following colors: green = OK, red = sensor problem, blue = Bluetooth problem, and yellow = Ethernet problem. Default value is 0 (Off).
commands/object/soundfile	Plays a sound by sound file name. For example, <code>{"data": "(1) Power On.wav"}</code> . See <a href="#">commands/object/soundinfo</a> .
commands/object/volume	Sets the speaker volume. Range: 0 to 100. Default value is 75%.
commands/object/soundRepeat	Sets the number of times a sound is played. Default value is 1.
commands/object/soundinfo	Lists the names and index numbers of all the sound files and their total size (in bytes). No JSON content needs to be entered. Simply enter 0 and then publish.
commands/object/ioChannel1	Sets the xP1 universal point value, if applicable.
commands/object/ioChannel2	Sets the xP2 universal point value, if applicable.

## Metrics Topics

Publish to a metrics topic using the format `commands/metrics/SUBTOPIC` and write to the `data` JSON key.

TABLE 5: METRICS PUBLISH TOPICS

Topic	Description
<code>commands/metrics/clear</code>	Deletes all metrics data in the database. Set to 1 (On) or 0 (Off). Default value is 0 (Off).
<code>commands/metrics/compute</code>	Computes metrics now (forces update). Set to 1 (On) or 0 (Off). Default value is 0 (Off). If there isn't enough data, the calculation isn't executed.
<code>commands/metrics/history</code>	Retrieves all metrics data (10,080 samples from the past 30 days). Set to 1 (On) or 0 (Off). Default value is 0 (Off).

## Firmware and Reboot Topics

TABLE 6: FIRMWARE AND REBOOT PUBLISH TOPICS

Topic	Description
<code>commands/fw/updateFw</code>	Initiates a firmware upgrade. Indicate the firmware version using the format <code>{"data": {"version": "<i>FIRMWARE_VERSION</i>"}}</code> .
<code>commands/reboot</code>	Reboots the O3 using the following message: <code>{"data": {"source": "enteliWeb", "message": "command coldstart"}}</code> .

## Setting Topics

If you are connected to the O3's internal MQTT broker, you can directly modify the O3's configuration file.

Modify a setting using the format `setting/TOPIC/SUBTOPIC` and write to the `data` JSON key. The change is then applied to `config/TOPIC/SUBTOPIC`.

For example, to change the unit for temperature from the default Celsius (°C) to Fahrenheit (°F), publish `{"data": "F"}` to `setting/units/temp`. To change it back to Celsius, publish `{"data": "C"}`. (Do not include the degree symbol °)

TABLE 7: SETTING TOPICS

Topic	Description
setting/bacnet/eth/Enable	Enables BACnet/Ethernet when set to true. Default value is true.
setting/bacnet/bnip/Enable	Enables BACnet/IP when set to true. Default value is false.
setting/bacnet/bnip/IpMode	<i>Not currently supported.</i> The topic <code>config/bacnet/bnip/IpMode</code> is read-only. To set up the O3 as a foreign device, you must use a BACnet front-end application. For more details, see the <i>O3 Edge BACnet Application Guide</i> on <a href="https://support.o3hub.com">support.o3hub.com</a> .
setting/bacnet/bnip/udpPort	Sets the BACnet/IP UDP port. Default value is 47808.
setting/bacnet/device_name	Sets the BACnet device name.
setting/bacnet/id	Sets the BACnet device ID. Must be a unique value between 1 and 4194302. Default value is based on Ethernet MAC address.
setting/ble/enable	Enables the Bluetooth LE communication when set to true. Default value is true.
setting/ble/pin	Sets the 6-digit authentication code for read/write access via Bluetooth. Default value is 000000. <a href="#">Reboot the device</a> after changing the PIN.
setting/ble/txPower	Sets the Bluetooth maximum transmit power. Must be a value between 1 and 8, corresponding to the following states: -40, -20, -16, -12, -8, -4, 0, 4 dBm.

TABLE 7: SETTING TOPICS

Topic	Description
setting/fw/urlFw	Sets the location of the server where firmware upgrades are stored.
setting/io/1/setup	Sets the xP1 universal point type (analog-input, analog-output, binary-input, binary-output) and configuration (0-5-volt, 0-10-volt, 10kohm-thermistor). For example, <code>{"data": {"type": "analog-input", "option": "0-5-volt"}}</code> .
setting/io/2/setup	Sets the xP2 universal point type (analog-input, analog-output, binary-input, binary-output) and configuration (0-5-volt, 0-10-volt, 10kohm-thermistor). For example, <code>{"data": {"type": "analog-output", "option": "0-10-volt"}}</code> .
setting/mqttPassword	Sets the password to access the internal MQTT broker. Default value is admin. <i>This password must be changed after initial login to prevent unauthorized use of the broker.</i> The password is case-sensitive and must be 5 to 31 characters in length. Special characters are allowed. <a href="#">Reboot the device</a> after changing the password.
setting/mqttPort	Sets the port used by the internal MQTT broker. Default value is 1883 (TCP).
setting/NTP	Enables Network Time Protocol service when set to true, causing the O3 to display UTC time. Default value is true.
setting/units/light	Sets the illuminance unit of measurement: lx (lux) or ft-c (foot-candle). Default value is lx.
setting/units/temp	Sets the temperature unit of measurement: C (Celsius) or F (Fahrenheit). Default value is C.

# BACnet Secure Connect

The O3 Edge allows you to configure and manage your Secure Connect connections using MQTT.

## Setting up Secure Connect using MQTT

The O3 currently supports BACnet Secure Connect (BACnet/SC) with a Delta enteliWEB/enteliCLOUD Primary Hub running in auto-config mode, to allow automatic certificate management.

To configure the O3, you must first create a Primary Hub on your site, or select an existing Primary Hub on enteliWEB, after which you can create a Secure Connect connection. For more information about creating a Secure Connect primary hub, see [Creating a Secure Connect Hub \(Primary\)](#).

Once you have a BACnet Secure Connect hub running in enteliWEB or enteliCLOUD, you can then configure and approve the device in MQTT using the following steps:

1. [Set the Auto Config URL](#)
2. [Enable Secure Connect](#)
3. [Approve the connection](#)
4. [Check connection status](#)

## Set the Auto Config URL

1. On your enteliWEB or enteliCLOUD site, click the **Secure Connect Primary Hubs** tab, and select your primary hub.
2. Select and copy the Auto Config URL.
3. Using MQTT, publish your Auto Config URL, e.g. `autoConfigUrl = {"data": "wss://192.168.0.115.56321"}`, to the topic `setting/bacnet/sc/autoConfigUrl`.

You can now enable the Secure Connect connection.

## Enable Secure Connect connection

- To enable Secure Connect, publish to the topic `setting/bacnet/sc/Enable` and write to the `data` JSON key, i.e. `{"data": "true"}`

You can now approve the device in enteliWEB.

## Approve device in enteliWEB

- From the **Secure Connect Primary Hubs** tab, select the pending device in your approval queue, and click **Approve**.

The approved device will join the BACnet/SC network and appear in the corresponding site in enteliWEB.

## Check connection status

The following topics can be used to check the status of your Secure Connect connection.

### General Connection Status:

```
events/object/scConnectionStatus
```

### Primary Hub Status

```
events/object/scPrimaryHubStatus
```

### Failover Hub Status

```
events/object/scFailoverHubStatus
```

## Disable Secure Connect Connection

To disable your Secure Connect connection:

- Using MQTT, publish to the topic `setting/bacnet/sc/Enable` and write to the `data` JSON key, e.g. `{"data": "false"}`.

The Secure Connect connection is disabled, and the O3 will no longer communicate on the BACnet/SC network.

# Using MQTT with EnOcean Devices

The O3 MQTT broker allows access to read and write EnOcean-related BACnet objects. The O3 Edge has a variety of MQTT topics to configure, monitor and program the device.

## Registering Objects

Before you can read and write BACnet objects on MQTT, you must first register them. This tells the O3 that you want to receive MQTT messages when these objects change, and configures it to send them.

The following BACnet objects are supported for EnOcean devices: AV, BV, MV, and CSV.

You can register EnOcean-supported analog value objects such as AV301001 to display sensor readings, or binary value objects such as BV301998 to command the Learn Mode object to on/off.

For example, if you have a toggle switch device, you must first register the Device Address CSV object.

To register properties, publish [ `"CSV901999"` ] to `commands/object/register`.

To register more than one object publish the JSON key in the following format: [ `"CSV901999"`, `"CSV301996"`, `"BV301998"` ].

## Reading Objects

- To read properties, subscribe using the following format, e.g. `events/object/BV301007`, to `events/object/ObjectInstance`. The object instance return the present value as a default property.

## Writing Objects

- To write objects, publish using the following format, e.g. `{"data": "77.7"}`, and set a present value to `commands/object/ObjectInstance`.

# Revision History

Edition	Date	Description
1.0	October 2021	New document based on <i>03 Edge 2.0 MQTT API Reference Guide</i> .