RADIX™

WHITE PAPER

MQTT SPARKPLUG

OVERVIEW

Radix-IoT has added a new MQTT data source to the Mango OS arsenal that implements the Sparkplug specification. Sparkplug is an open-source software specification that provides MQTT clients the framework to seamlessly integrate data from their applications, sensors devices and gateways within the MQTT infrastructure. The MQTT Sparkplug specification has gained rapid industry adoption since its launch due to its decoupled architecture, lightweight report by exception and simplicity. MQTT Sparkplug is already playing a leading role in the Industrial Internet of Things (IIoT) protocol space.

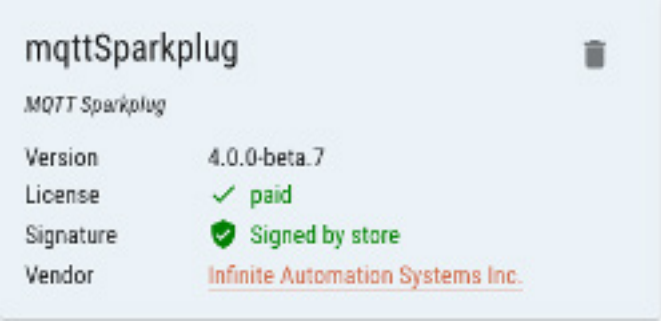
WHY SPARKPLUG?

While MQTT offers scalability, security and flexibility the protocol does not specify any standards in topic namespaces, payload & data structures or state management. Any application that intends on consuming the MQTT data will need to accommodate a plethora of different variants. Sparkplug was designed to address this issue by eliminating the potential for inconsistent MQTT implementation and creating a true plug and play MQTT environment. The Specification addresses the following components withing an MQTT infrastructure:

- **Defined MQTT Topic Namespace:** All clients and hosts in the system settle on a common namespace format, creating a vendor-neutral environment and removing the responsibility from the system administrator to maintain a strict namespace convention.
- **Defined MQTT Payload:** The original MQTT specification did not specify any details on the payload, making it very difficult for SCADA and IoT platforms to create a vendor neutral data receiver. The sparkplug specification went beyond just defining the payload and included all the information required to create a digital twin of the edge device in the payload specification.
- **Defined MQTT State Management:** MQTT brokers are aware of the state of each client connected to it by using the specifications birth and death payloads. This brings a number of advantages such as: Edge devices can be configured to only publish data when the primary host is connected to the broker reducing unnecessary messages and saving battery power and bandwidth

MANGO-OS MQTT SPARKPLUG DATA SOURCE

The new MQTT Sparkplug data source is available on Mango OS v4 and comes pre-installed on the Mango Enterprise package or can be installed through the Mango OS modules dashboard. The module does not require any additional licensing to use. If the following component is not visible on your module's dashboard, you can install it by clicking the "Check for upgrades" button, select the modules and click the "Upgrade/Install" button.



GETTING CONNECTED

MQTT Sparkplug uses the exact same network architecture as a standard MQTT system and at this stage the specification is only available on MQTT v3. Therefore, to use this data source you will need to have a MQTT v3 Broker that has been configured and is ready to connect to. There are a number of free MQTT brokers available. The most popular of them being Mosquitto.

Once the module is installed and you have a broker available you are now ready to create a MQTT Sparkplug Data Source. Navigate to the Data Sources dashboard and select "Create data source". Open the Data source type drop down and select MQTT Sparkplug.

Broker URL: The host name or IP address of the broker. The data source allows two different connection schemes, TCP or SSL. If TCP is chosen a plain TCP connection will be established with the broker, if SSL is chosen a secure TLS connection will be established. The connection scheme is chosen by prefixing the URL with either tcp or ssl. Eg:

- tcp://localhost:1883
- ssl://localhost:8883

The port that the broker is listening on should be included in the URL as shown above.

Client ID: Each client that connects to a MQTT v3 Broker requires a unique Client ID across the entire Broker. This ID is used by the broker to store data related to the client and should always remain the same.

SCADA Host ID: Sparkplug specifies that any application host node that will subscribe and publish Sparkplug messages must have a SCADA Host ID.

Primary SCADA: In a typical SCADA/IIoT system there will only be one primary SCADA Host node. Edge node devices can be configured to only publish data when a primary host is connected.

X509 CA: If the SSL connection scheme has been used, you will need to provide a CA certificate. The certificate should be pasted in as plain text.

Topic Filters: The data source can be configured to only subscribe to a sub-set of all topics.

Auto reconnect: Sets whether the client will automatically attempt to reconnect to the server if the connection is lost.

- If set to false, the client will not attempt to automatically reconnect to the server in the event that the connection is lost.
- If set to true, in the event that the connection is lost, the client will attempt to reconnect to the server. It will initially wait 1 second before it attempts to reconnect, for every failed reconnect attempt, the delay will double until it is at 2 minutes at which point the delay will stay at 2 minutes.

Copy Rebirth Permissions: The data source allows you to choose whether all data points from a specific node should have the same permissions scheme as the Node Control/Rebirth data point.

QoS Type: Sets whether the client and server should remember state across restarts and reconnects.

Keep Alive interval: This value, measured in seconds, defines the maximum time interval between messages sent or received. It enables the client to

detect if the server is no longer available, without having to wait for the TCP/IP timeout. The client will ensure that at least one message travels across the network within each keep alive period. In the absence of a data-related message during the time period, the client sends a very small “ping” message, which the server will acknowledge. A value of 0 disables keepalive processing in the client.

The default value is 60 seconds.

Connection Timeout: Sets the connection timeout value. This value, measured in seconds, defines the maximum time interval the client will wait for the network connection to the MQTT server to be established. The default timeout is 30 seconds. A value of 0 disables timeout processing meaning the client will wait until the network connection is made successfully or fails.

The default value is 30 seconds.

Username: The username of the credentials that will be used to connect to the Broker.

Password: The password of the credentials that will be used to connect to the Broker.

ADDING EDGE NODE DEVICES

The Data Source provides a simple and intuitive method to add and rebirth edge nodes. To add an Edge Node, all that you will need is the group ID and Edge Node ID of the device that you are adding. Once you have inserted this information you will need to configure the permissions of the device then click add. The data source will add a Binary rebirth point according to the specification and publish a rebirth request to the device. When the device receives the request, it will publish a Sparkplug BIRTH message. The BIRTH message contains all the information required to create the data points of the device. These data points will automatically be created and will be viewable on the data points tab. If the Edge Node devices has a sub device connected to it, these data points will automatically be created as well. If the MQTT Sparkplug data source has been configured to copy rebirth permissions, each point will have the same permissions as when the point is added. A manual Rebirth request can also be sent to devices that have already been added.

Published by
Radix IoT 2021

Radix IoT, LLC
14555 N. Dallas Parkway #125
Dallas, TX 75254
United States

For more information, please contact
our Customer Support Center.
(Charges depending on provider)
Email: info@radixiot.com

RadixIoT.com