SignalFire Ranger – Ignition Integration Guide

The SignalFire Ranger is an Internet of Things (IoT) cellular LTE-M telemetry node capable of communicating to a variety SCADA host software using Message Queuing Telemetry Transport (MQTT) and Sparkplug protocol. This guide will walkthrough the steps necessary in connecting the SignalFire Ranger to Ignition from Inductive Automation.

Setup MQTT Broker
The SignalFire Ranger connects to an MQTT broker to publish its measurement data. Ignition must also connect to the same MQTT broker to subscribe to the measurements published by the Ranger and other nodes. This requires having access to your own MQTT broker and creating credentials for the Ignition server and the Ranger node to use to connect and login to the broker.

These MQTT broker credentials include:
- Hostname or IP Address (required)
- Port number (required)
- Username/Password (optional)
- TLS Settings (optional): CA certificate for host validation, Client certificate and key

The links below contain some options for setting up your own MQTT broker.
CloudMQTT – Hosted MQTT brokers
Mosquitto, EMQ X, RabbitMQ – Open-source MQTT brokers for Windows, Mac, or Linux
HiveMQ, Chariot, MQTT Distributor – Enterprise MQTT brokers
Install Ignition
Follow the steps in this Ignition installation guide to install Ignition on Windows, Mac, or Linux.

After Ignition is installed and the commissioning process is completed, confirm that you can login to the Ignition Gateway webpage.

Figure 1. Ignition Gateway webpage
Install MQTT Engine Ignition Module

Ignition uses the MQTT Engine module, a third-party module from Cirrus-Link, to connect to an MQTT broker and act as a Sparkplug SCADA host for Sparkplug Edge of Network (EoN) nodes publishing data to the broker.

Download and install the Cirrus-Link MQTT Engine Module for your version of Ignition, and follow the instructions in this Ignition Module installation guide to install the module. This video walkthrough may be helpful as well.

Figure 2. Ignition MQTT Engine Module Settings
Configure MQTT Engine
The MQTT Engine Module must be configured with credentials to connect to your MQTT broker and start receiving data from Ranger nodes.

For this example, the following MQTT credentials will be used:
**Hostname:** mosquitto.signal-fire.cloud

**Port:** 1883

**Username:** demo

**Password:** password

First, select the “Servers” tab, and click “Create new MQTT Server Setting...”. Next, give this server connection a name, and enter the URL, Username, Password, and optional TLS credentials. Click the “Create New MQTT Server Setting” button at the bottom of the page to create.

Figure 3. Ignition MQTT Server Settings
After adding the MQTT Server Setting, we can confirm that Ignition is connected to the MQTT broker.

**Figure 4. Ignition MQTT Server showing Status “Connected”**

**MQTT Engine Configuration:**
After installing the Cirrus Link MQTT Engine module on your Ignition server, and configuring the module to connect to your MQTT broker, there are a few important configuration settings that need to be considered for allowing the SignalFire Ranger to be added to Ignition’s tag provider as an Edge Node.

**Primary Host ID**
Sparkplug protocol allows for SCADA Hosts, such as Ignition, to publish their name, also known as their Host ID, as well as their Online or Offline state, so that Sparkplug EoN nodes, such as the SignalFire Ranger, can detect whether or not one or more SCADA hosts are online. The SignalFire Ranger requires, by default, at least one SCADA Host to be Online in order to publish data to the broker.

If this Ignition server is to be considered the Primary SCADA Host for the Sparkplug EoN nodes connected to the MQTT broker, then it is important to give the server a Primary Host ID. This is simply a name used for EoN nodes to identify this SCADA Host server from other servers, and must be unique across all SCADA Host servers connected to this MQTT broker. In this example, the Primary Host ID is set to “IgnitionTest”.
Figure 5. Ignition MQTT Engine Module set to Primary Host ID “IgnitionTest”

**Unblock Node and Device Commands**

By default, Ignition’s MQTT Engine Module blocks all commands to Sparkplug EoN node and devices. This effectively makes this Ignition server read-only, without the ability to send commands and write tags on the remote nodes. To allow for such commands, uncheck the “Block outbound edge node tag writes” and “Block outbound device tag writes” checkboxes.

Figure 6. Ignition MQTT Engine Module set to allow outbound edge node and device tag writes
Ignition Tags
Ignition’s MQTT Engine Module will automatically handle creating tags for the Sparkplug metrics being published to the MQTT broker by the Sparkplug EoN nodes, such as the SignalFire Ranger. It will also automatically publish tag write commands back to the respective Sparkplug EoN node when the tag is writing to by the Ignition server.

There are several ways to view Ignition’s tag database, including the tag browser built into the Ignition Gateway webpage available from the “Status” section. The tags created from Sparkplug EoN nodes’ published metrics are located in the “Edge Nodes” directory.

![Ignition Gateway webpage tag browser](image)

Figure 7. Ignition Gateway webpage tag browser
The Ignition Designer also allows for tag browsing using an expandable tree view.

Figure 8. Ignition Designer Tag Browser.