

MT: Configuration

MQTT Transmission provides a configuration section to the Ignition Gateway. These can be seen in the Configure section of the Ignition Gateway web UI in the left-hand navigation pane - Configure MQTT Transmission. There are two configuration pages "History" and "Settings".

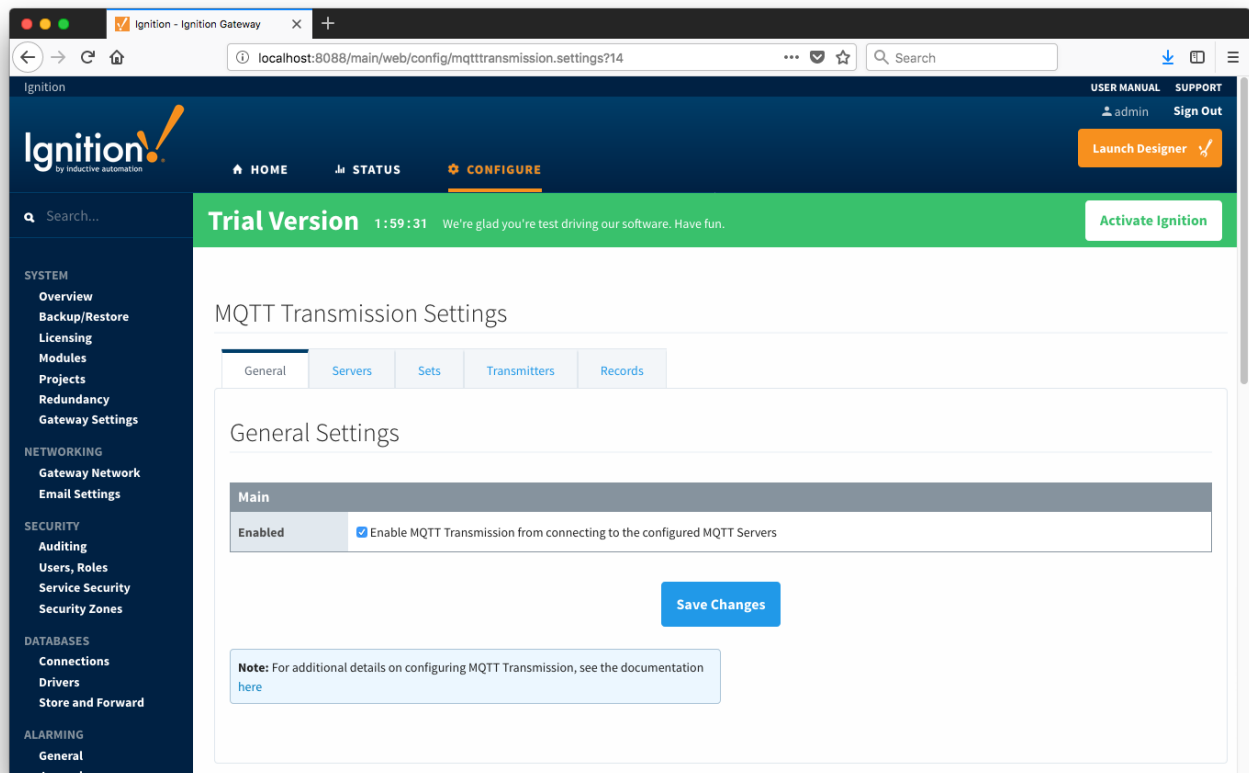
Settings

Once in the "Settings" configuration section there are five tabs: General, Servers, Sets, Transmitters, and Records. Each of these tabs is described in detail in the following sections.

General

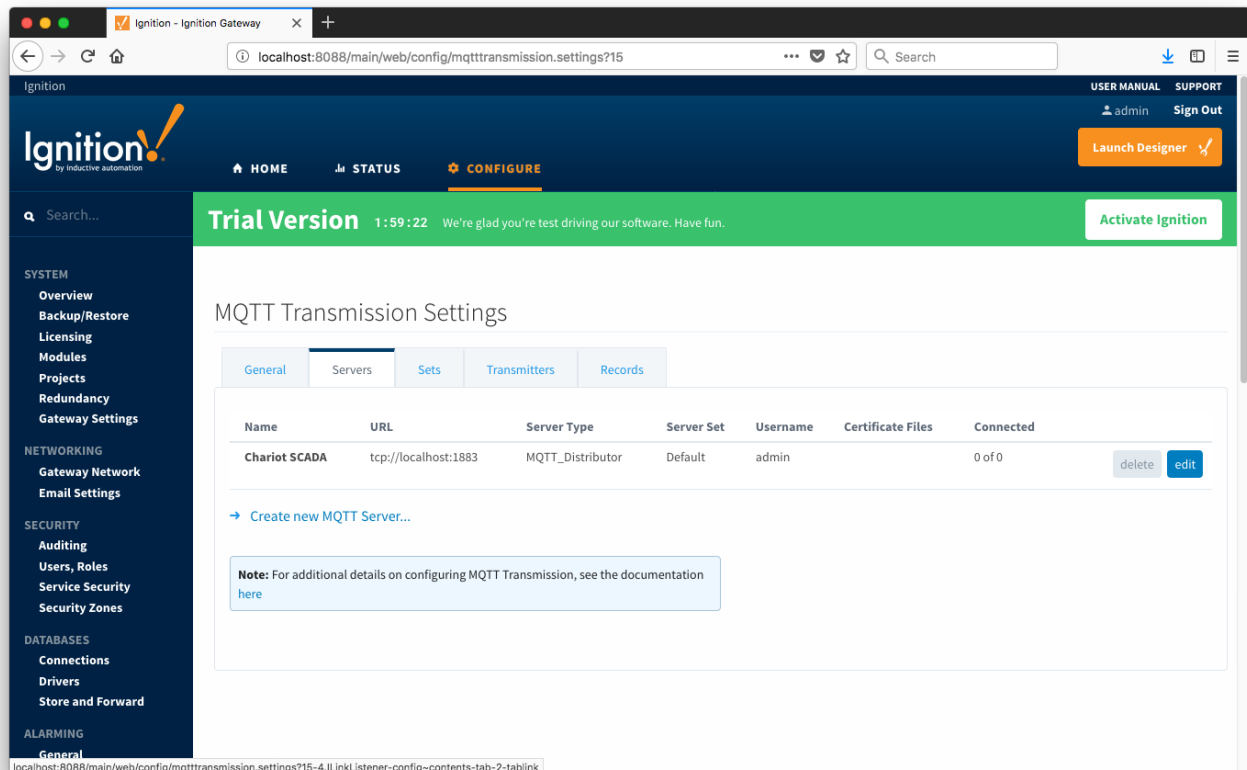
The General Settings tab contains advanced configuration options for the default MQTT Server settings created in the Servers tab. These are described below.

- **Enabled**
 - Whether or not to enable or disable MQTT Transmission from connecting to the configured MQTT Servers.



Servers

The first tab is a list of MQTT Servers that MQTT Transmission should connect to. By default, MQTT Transmission is configured to connect to a local MQTT Distributor based MQTT Server. It is set up to connect to localhost, port 1883, using the default username/password pair of admin/changeme. Out of the box MQTT Transmission will work with MQTT Distributor and its default configuration.



Additional or alternative MQTT Servers can be configured in MQTT Transmission. Often times more than one will be configured to handle fail-over or in redundant or geographically distributed systems. The configuration options for servers are listed below.

Main Properties

- **Name**
 - This is the friendly name of the MQTT Server used to easily identify it
- **URL**
 - This is the URL of the MQTT server. Its format is as follows: [protocol]://[location]:[port]. Each of these are shown below
 - protocol - Either tcp or ssl
 - location - The server location. e.g. localhost, [myserver.chariot.io](#), [mydomain.com](#), etc
 - port - The port the MQTT Server is listening on. Generally this is 1883 if using TCP or 8883 if using SSL
- **Server Type**
 - This is the type of MQTT Server to connect to
 - Chariot - If connecting to a Cirrus Link Chariot on-premise or Chariot cloud based MQTT Server
 - MQTT_Distributor - If connecting to a Ignition MQTT Distributor server
 - Third_Party - If connecting to a third party 3.1.1 compliant MQTT Server
 - AWS_Greengrass - If connecting to an AWS Greengrass Server.
 - Note: This server type should not be used when using MQTT Transmission in conjunction with MQTT Engine.
- **Server Set**
 - The Set that this server is a member of.
- **Username**
 - The MQTT username to use in the MQTT connect packet. This is required if the MQTT Server to connect to requires it.
- **Password**
 - The MQTT password to use in the MQTT connect packet. This is required if the MQTT Server to connect to requires it.

TLS Properties

- The configuration for TLS changed between MQTT Transmission v3.4.6 and v3.4.7. See this document for TLS configuration: [Configuring Secure MQTT Communication](#)

Advanced Properties

- **Client ID**
 - Optional MQTT client ID to use. If left blank, a random client ID will be created of the form 'MT-xxxxxxx-xxxx-xxxx'. **This should almost always be left blank. Do not use this setting unless it is an absolute requirement that a specific client ID be used for connection to the MQTT server.**

- **Keep Alive**
 - The MQTT client keep alive time (in seconds).
- **Random Startup Delay**
 - The Random Startup Delay in milliseconds of the form 'min-max' where min is the low end and max is the high end of the random range. This will provide a random delay in the startup and connection of the MQTT Client with the MQTT Server. This is useful in use cases where large numbers of clients might be reconnecting simultaneously to the MQTT Server. By introducing a randomized delay, the rate at which clients will connect will be spread out over a period of time, reducing the load on the server.
 - Example: 1000-5000

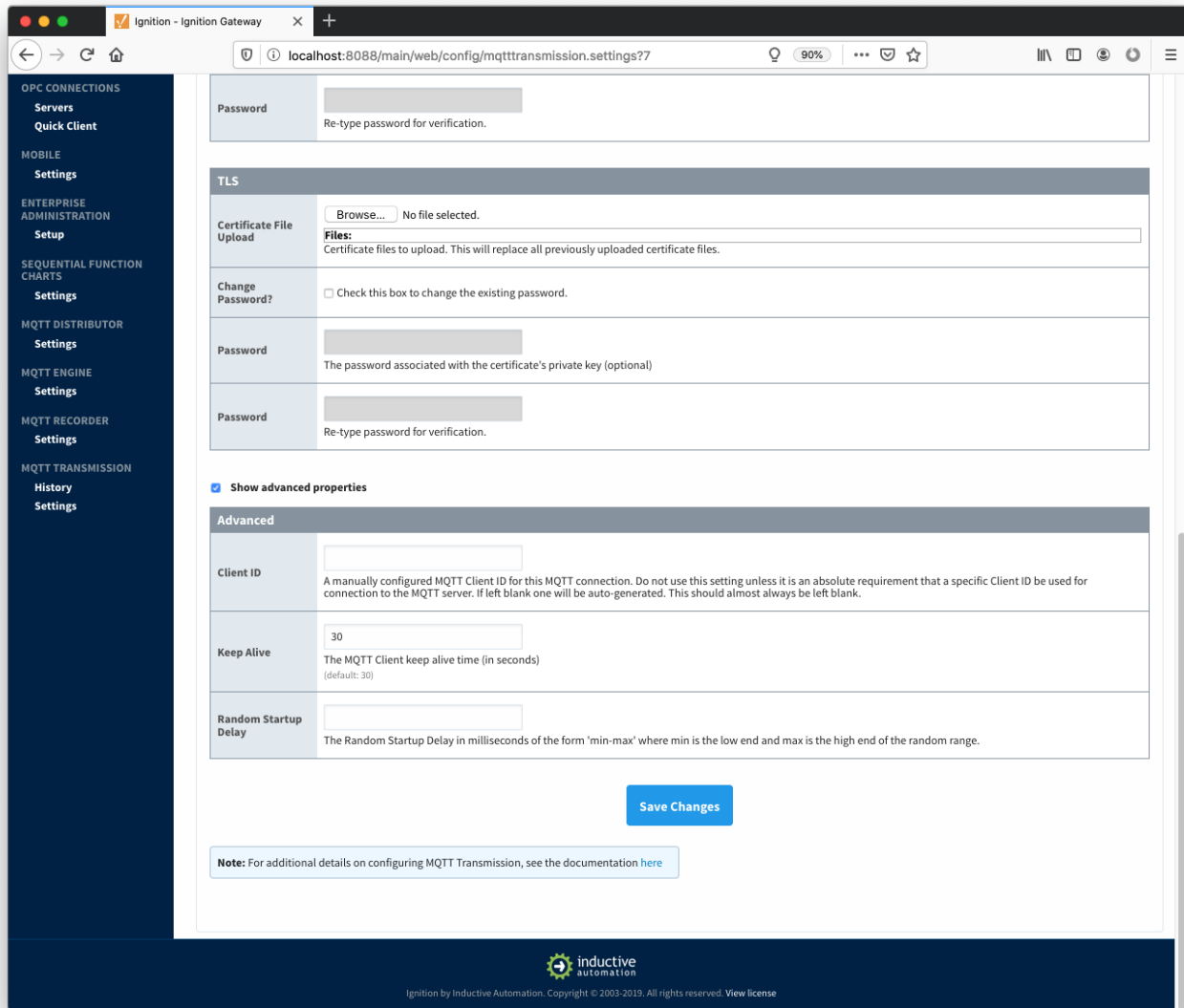
Clicking on the 'Create new MQTT Server...' link will bring up the following form to add a new Server.

The screenshot shows the Ignition Gateway web interface. The browser address bar indicates the URL is `localhost:8088/main/web/config/mqtttransmission.settings?14`. The interface has a dark blue header with the Ignition logo and navigation links: HOME, STATUS, and CONFIGURE. A green banner at the top indicates it's a 'Trial Version' with a timer at 1:51:56 and an 'Activate Ignition' button. On the right, there are links for 'USER MANUAL', 'SUPPORT', 'admin', 'Sign Out', and a 'Launch Designer' button.

A left sidebar contains a search bar and a tree view of system components: SYSTEM (Overview, Backup/Restore, Licensing, Modules, Projects, Redundancy, Gateway Settings), NETWORKING (Gateway Network, Email Settings), SECURITY (Auditing, Users, Roles, Service Security, Security Zones), DATABASES (Connections, Drivers, Store and Forward), ALARMING (General, Journal, Notification, On-Call Rosters, Schedules), TAGS (History, Realtime), OPC-UA SERVER (Certificates, Devices, Settings), OPC CONNECTIONS (Servers, Quick Client), and MOBILE.

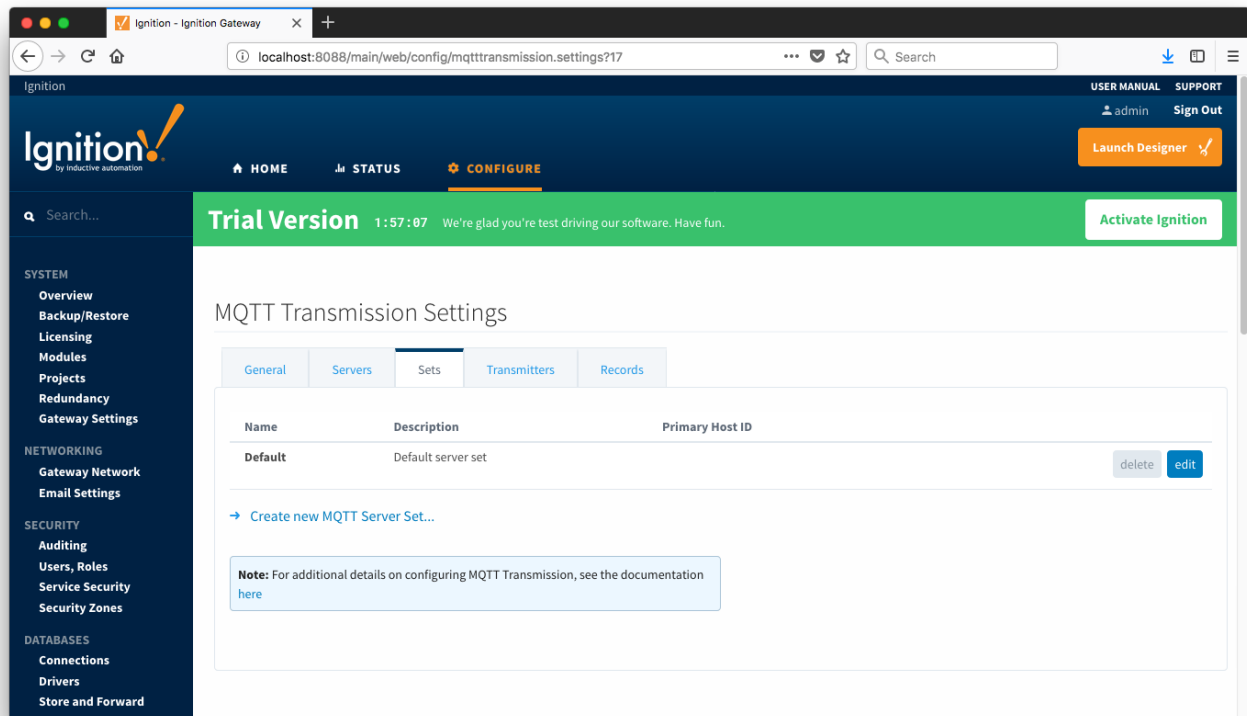
The main content area is titled 'MQTT Transmission Settings' and has tabs for General, Servers, Sets, Transmitters, and Records. The 'Servers' tab is active, showing the 'Edit MQTT Server' form. The form has a 'Main' section with the following fields:

Name	Chariot SCADA <small>The friendly name of this MQTT Server</small>
URL	tcp://localhost:1883 <small>The URL of the MQTT Server to connect to. Should be of the form tcp://mydomain.com:1883 or ssl://mydomain.com:8883</small>
Server Type	MQTT_Distributor <small>The type of MQTT Server to connect to (default: MQTT_Distributor)</small>
Server Set	Default <small>The Server Set this MQTT Server is associated with</small>
Username	admin <small>The username for this MQTT connection if required by the MQTT Server (optional)</small>
Change Password?	<input type="checkbox"/> Check this box to change the existing password.
Password	<input type="password"/> <small>The password for this MQTT connection if required by the MQTT Server (optional)</small>
Password	<input type="password"/> <small>Re-type password for verification.</small>



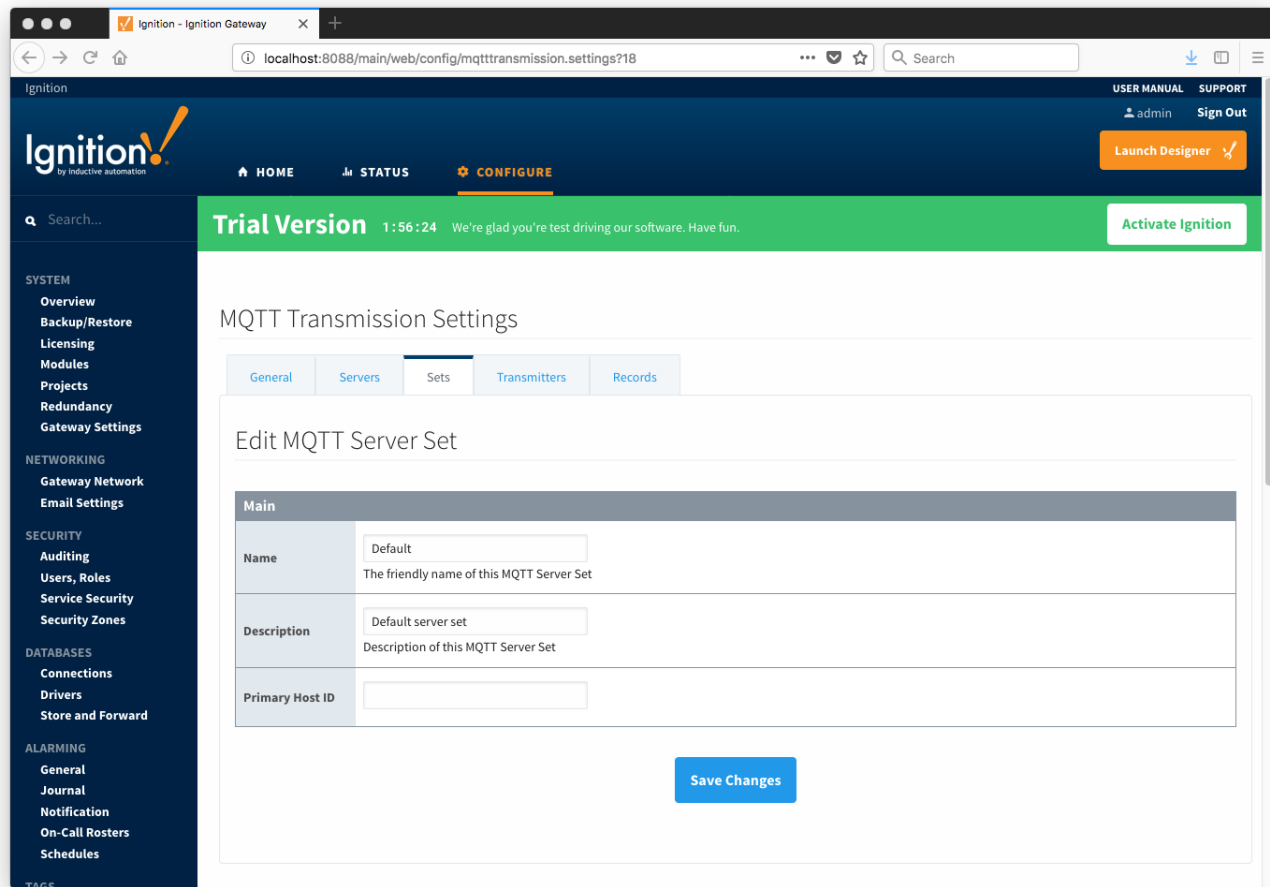
Sets

The Sets tab contains a list of server sets. Each set represents a logical grouping of MQTT servers. When a set is referenced by a Transmitter configuration a single connection to one of the servers in the set will be maintained. The other servers will act as failover in the case that a connection with the first is lost. Server sets are disjoint, meaning that a single MQTT server cannot be in more than one set.



Additional sets can be configured in MQTT Transmission for situations where multiple Transmitters need to point to different sets (or groupings) of MQTT servers. The configuration options for sets are listed below.

- **Name**
 - This is the friendly name of the set used to easily identify it.
- **Description**
 - This is a friendly description of the set.
- **Primary Host ID**
 - The primary host ID to use for 'state' notifications. These notifications are used to notify MQTT Transmission if the primary backend application loses connection with the MQTT Server, meaning it has gone 'offline'. If MQTT Transmission is notified that the primary backend application has gone 'offline', it will close it's client connection with the MQTT server and walk to the next MQTT server defined in the set. If the primary host ID is not set, MQTT Transmission will not subscribe on the notification topics and not receive any 'state' notifications.
 - This must contain only letters, numbers, and the following special characters: . \$ % @ ! - _ ^ *



Transmitters

Transmitters are the agents within MQTT Transmission that monitor tags, convert them to Sparkplug Messages, and publish them to an MQTT Server. Each transmitter is configured with a server Set and will attempt to maintain an MQTT client connection with one server in that Set at all times.

There are two types of Transmitters: Default and Custom. These types are explained in more detail below.

Default Transmitter

MQTT Transmission provides a single Default Transmitter. The Tags that this Transmitter monitors must be placed in the "Edge Nodes" folder within the configured Tag Provider. Their Tag paths must conform to the strict structure of:

```
<Provider>/Edge Nodes/<GroupID>/<EdgeNodeID>/<DeviceID>
```

The individual elements of the above tag path (Group ID, Edge Node ID, and Device ID) are used when building up the MQTT message payload that will represent the Tags.

The configuration options for the Default Transmitter are listed below.:

- **Name**
 - Default Transmitter
- **Enabled**
 - Whether the Default Transmitter is enabled
- **Tag Provider Name**
 - The name of the tag provider that Transmission will monitor. By default this is the Ignition 'default' provider.
- **Tag Pacing Period**
 - The buffer period for outgoing Transmission messages in milliseconds. The default is 1000ms. This means when a tag change event is detected 1000ms will elapse before an MQTT message is sent. This allows additional tag change events to be buffered and put into the message and in turn reduce the number of generated MQTT messages.
- **Set**
 - The Set that the default MQTT Transmission client will connect to.
- **Discovery Delay**
 - An optional startup delay, in milliseconds, to wait before scanning for Tags to monitor. This is useful when Tags are dynamically created on initial startup, as is the case when using the MQTT Engine module.

- **History Store**
 - The MQTT Transmission History Store to use with the Default Transmitter.
- **In-Order History**
 - Whether MQTT Transmission will publish historical messages in-order upon reconnecting to an MQTT Server before resuming the publishing of live events.
- **Aliased Tags**
 - Whether to use aliases for tag names when published data messages as tag values change. This is used to optimize payload size when publishing data.
- **Compression**
 - The algorithm to use to compress payloads before they are published to the MQTT Server. If 'NONE' is selected then compression is disabled.
- **Block Commands**
 - Whether to block commands and tag writes received as messages from the MQTT server.
- **Convert UDTs**
 - Whether to convert UDT members to normal Tags before publishing. If enabled the Tags representing the UDT member will retain their member path prefixed by the UDT Instance name.
- **Publish UDT Definitions**
 - Whether to publish UDT definitions in the birth messages that are publish upon every connection or reconnection with an MQTT server.

MQTT Transmission Settings

General Servers Sets Transmitters Records

Default Custom

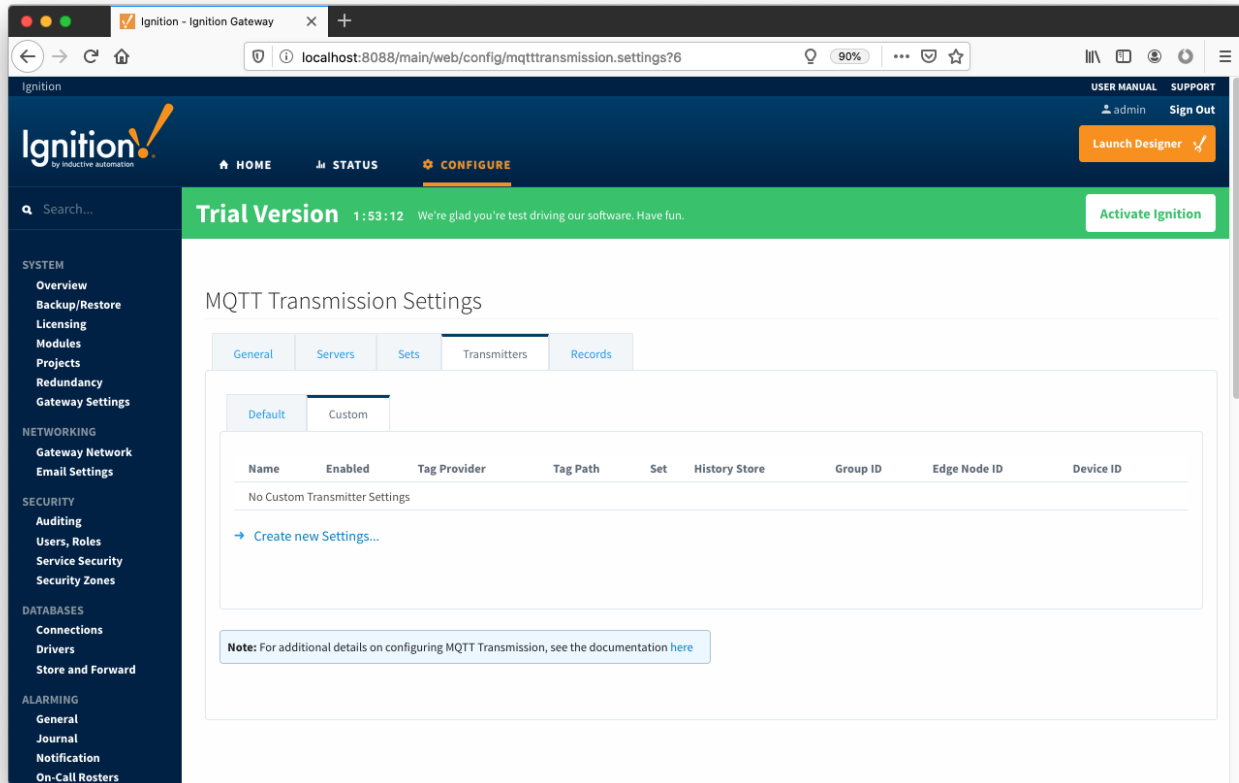
Configuration

Name	Default Transmitter Name for the Default Transmitter
Enabled	<input checked="" type="checkbox"/> Enable Default Transmitter
Tag Provider Name	default The Name of the tag provider
Tag Pacing Period	1000 The waiting period in milliseconds after an initial tag change event before publishing all changed tags
Set	Default The MQTT Server Set to use with the Default Transmitter
Discovery Delay	0 The Discovery Delay in milliseconds. This is useful when using MQTT Engine as the tag provider (default: 0)
History Store	- none - The MQTT Transmission History Store to use with the Default Transmitter
In-Order History	<input type="checkbox"/> Flush history in-order (synchronously) before live data resumes (default: false)
Aliased Tags	<input type="checkbox"/> Use aliases for tag names to optimize payload sizes when publishing data
Compression	NONE The algorithm to use for compressing payloads before publishing
Block Commands	<input type="checkbox"/> Block incoming commands (writes) to Edge Node and Device Tags
Convert UDTs	<input checked="" type="checkbox"/> Converts UDT members to normal Tags before publishing
Publish UDT Definitions	<input checked="" type="checkbox"/> Publish UDT Definitions in BIRTH

Save Changes

Custom Transmitters

Custom Transmitters behave in much the same way as the Default Transmitter. However, unlike the Default Transmitter, Custom Transmitters are not bound to a specific "Edge Nodes" folder and do not have any required folder path structure. The Group ID, Edge Node ID and Device ID are explicitly defined in a Custom Transmitter configuration instead of inferred from the tag path, as in the Default Transmitter. This allows Custom Transmitters to point at any folder in any provider and monitor the contained tags and/or tag folders.



Each Custom Setting has the following fields

Tag Settings

- **Name**
 - The name of the Custom Transmitter Setting.
- **Enabled**
 - Whether the Default Transmitter is enabled
- **Tag Provider Name**
 - The name of the tag provider that Transmission will monitor.
- **Tag Path**
 - An optional folder path under the tag provider where the root folder of the tags can be found.
- **Tag Pacing Period**
 - The buffer period for outgoing Transmission messages in milliseconds.
- **Set**
 - The Set that the Custom Settings client will connect to.
- **Discovery Delay**
 - An optional startup delay, in milliseconds, to wait before scanning for Tags to monitor. This is useful when Tags are dynamically created on initial startup, as is the case when using the MQTT Engine module.
- **History Store**
 - The MQTT Transmission History Store to use with the Default Transmitter.
- **In-Order History**
 - Whether MQTT Transmission will publish historical messages in-order upon reconnecting to an MQTT Server before resuming the publishing of live events
- **Aliased Tags**
 - Whether to use aliases for tag names when published data messages as tag values change. This is used to optimize payload size when publishing data.
- **Compression**
 - The algorithm to use to compress payloads before they are published to the MQTT Server. If 'NONE' is selected then compression is disabled.
- **Block Commands**
 - Whether to block commands and tag writes received as messages from the MQTT server.
- **Convert UDTs**

- Whether to convert UDT members to normal Tags before publishing. If enabled the Tags representing the UDT member will retain their member path prefixed by the UDT Instance name.
- **Publish UDT Definitions**
 - Whether to publish UDT definitions in the birth messages that are publish upon every connection or reconnection with an MQTT server.

Sparkplug Settings

- **Group ID**
 - An ID representing a logical grouping of MQTT Edge Of Network (EoN) Nodes and Devices into the infrastructure.
- **Edge Node ID**
 - An ID that uniquely identifies the MQTT Edge Of Network (EoN) Node within the infrastructure.
- **Device ID**
 - An optional ID that uniquely identifies a Device within the infrastructure.

Note that if a 'Device ID' is not specified, any folder within the folder specified by the Tag Path will be considered a device folder and any Tags within it will be published as device Tags.

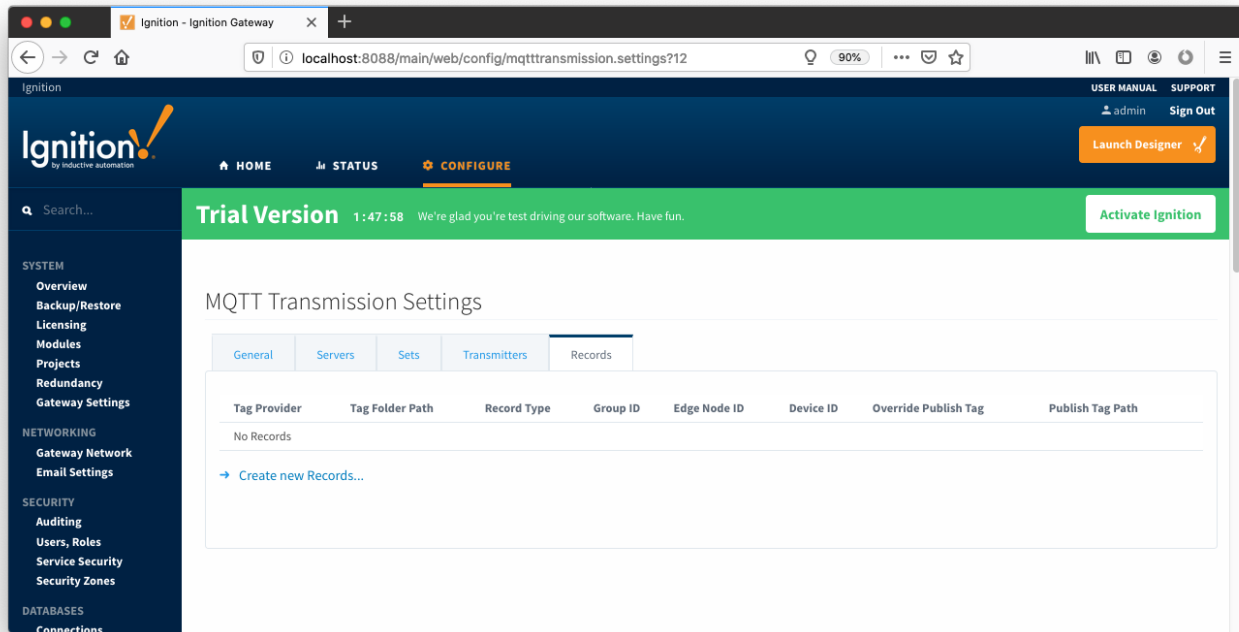
Clicking on the 'Create new Custom Settings..' link will bring up the following form to add a new Server.

The screenshot shows the Ignition Gateway web interface. The browser address bar indicates the URL is `localhost:8088/main/web/config/mqtttransmission.settings?7`. The interface has a dark blue header with the Ignition logo and navigation links: HOME, STATUS, and CONFIGURE. A green banner below the header indicates it's a 'Trial Version' with a timer at 1:51:52. The left sidebar contains a search bar and a menu with categories: SYSTEM, NETWORKING, SECURITY, DATABASES, ALARMING, TAGS, OPC-UA SERVER, OPC CONNECTIONS, MOBILE, and ENTERPRISE ADMINISTRATION. The main content area is titled 'MQTT Transmission Settings' and has tabs for General, Servers, Sets, Transmitters, and Records. The 'Sets' tab is active, showing a 'New Settings' form. The form has two tabs: 'Default' and 'Custom'. The 'Custom' tab is selected, and the form contains the following fields:

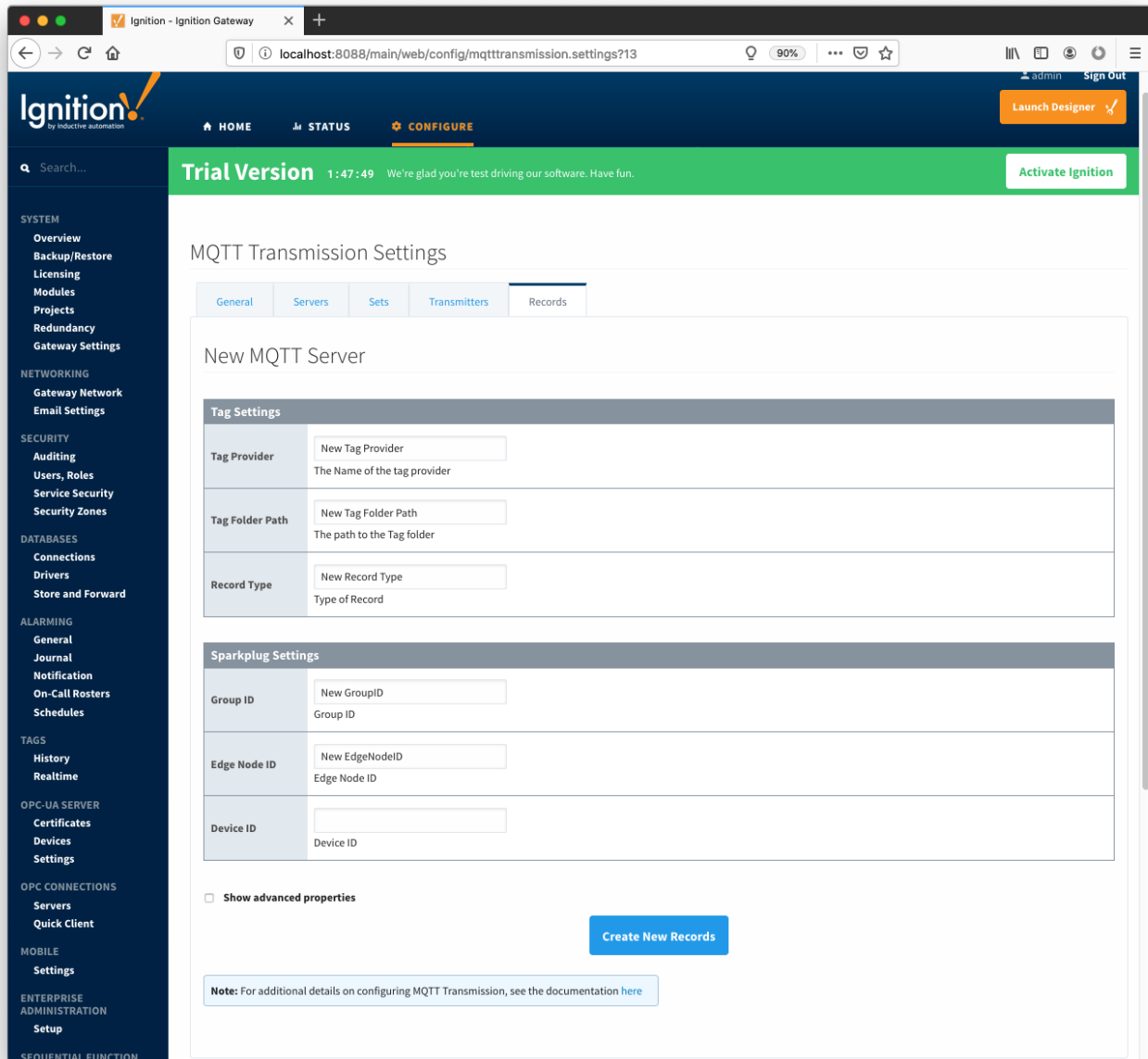
Tag Settings	
Name	<input type="text"/> A unique name for the Custom Transmitter
Enabled	<input checked="" type="checkbox"/> Enable Custom Transmitter
Tag Provider	<input type="text"/> The Name of the tag provider
Tag Path	<input type="text"/> A path to the root folder where the tag tree starts (optional)
Tag Pacing Period	<input type="text" value="1000"/> The waiting period in milliseconds after an initial tag change event before publishing all changed tags (default: 1,000)
Set	<input type="text" value="Default"/> The MQTT Server Set to use with this Transmitter
Discovery Delay	<input type="text" value="0"/> The Discovery Delay in milliseconds. This is useful when using MQTT Engine as the tag provider (default: 0)
History Store	<input type="text" value="- none -"/> The MQTT Transmission History Store to use with this Transmitter
In-Order History	<input type="checkbox"/> Flush history in-order (synchronously) before live data resumes (default: false)

Records

The 'Records' tab allows user to create a table of custom records. Each record defines a folder (under specified tag provider) in which user can create a number of tags that will be published as a record by clicking the 'Publish' checkbox. The 'Publish' tag will be created automatically in the folder specified.



Clicking on the 'Create new Records ...' link will bring up the following form to add a new Record:



Each custom record has the following fields under two categories: 'Tag Settings', and 'Sparkplug Settings'

Tag Settings

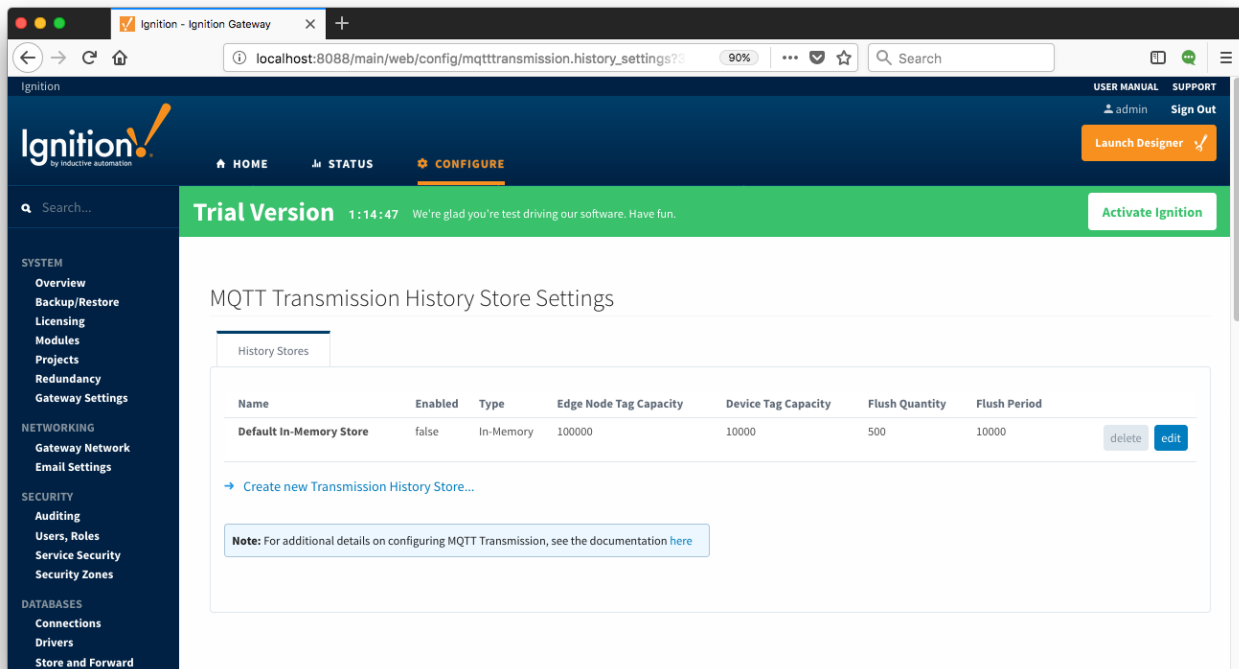
- **Tag Provider**
 - The name of the tag provider (i.e. default)
- **Tag Folder Path**
 - The path to the tag folder under specified tag provider
- **Record Type**
 - Type of Record. This can be any label or identifier.

Sparkplug Settings

- **Group ID**
 - An ID representing a logical grouping of MQTT Edge Of Network (EoN) Nodes and Devices into the infrastructure.
- **Edge Node ID**
 - An ID that uniquely identifies the MQTT Edge Of Network (EoN) Node within the infrastructure.
- **Device ID**
 - An optional ID that uniquely identifies a Device within the infrastructure.

History

The "History" page allows for the configuration of MQTT Transmission History Stores. In the event that a Transmitter loses its connection with the MQTT Server and is unable to reconnect, a History Store (if enabled) will store all messages corresponding to change events on the monitored tags. Once a connection with an MQTT server is reestablished the History Store will publish the stored messages with a flag set to indicate that the messages are "historical" to prevent confusion with live data values.



Each History Store has the following fields

- **Name**
 - The name of the History Store.
- **Enabled**
 - Whether the History Store is enabled.
- **Type**
 - The type of History Store.
- **Edge Node Tag Capacity**
 - The maximum number of edge node tags to store (per edge node) before dropping the oldest currently stored tag value.
- **Device Tag Capacity**
 - The maximum number of device tags to store (per device) before dropping the oldest currently stored tag value.
- **Flush Quantity**
 - The maximum number of tags tags to publish in a single message upon reestablishing communication.
- **Flush Period**
 - The period to wait in milliseconds between publishes when flushing messages upon reestablishing communication.

Ignition - Ignition Gateway

localhost:8088/main/web/config/mqtttransmission.history_settings?8

90%

Search

Ignition

USER MANUAL

SUPPORT

admin

Sign Out

Launch Designer

HOME

STATUS

CONFIGURE

Search...

SYSTEM

Overview

Backup/Restore

Licensing

Modules

Projects

Redundancy

Gateway Settings

NETWORKING

Gateway Network

Email Settings

SECURITY

Auditing

Users, Roles

Service Security

Security Zones

DATABASES

Connections

Drivers

Store and Forward

ALARMING

General

Journal

Notification

On-Call Rosters

Schedules

TAGS

History

Realtime

OPC-UA SERVER

Certificates

Devices

Settings

OPC CONNECTIONS

Servers

Quick Client

MOBILE

Settings

ENTERPRISE ADMINISTRATION

Setup

Trial Version

1:14:19

We're glad you're test driving our software. Have fun.

Activate Ignition

MQTT Transmission History Store Settings

History Stores

New Transmission History Store

Main

Name

New History Store

Name of this History Store

Enabled

☒ Enable this History Store

Type

Choose One

The Type of this History Store

Edge Node Tag Capacity

Maximum number of edge node tags to store (per edge node) before dropping oldest historical edge node tags

Device Tag Capacity

Maximum number of device tags to store (per device) before dropping oldest historical device tags

Flush Quantity

Upon reestablishing communication, the maximum number of tags tags to publish at a time

Flush Period

Upon reestablishing communication, the period to wait in milliseconds between publishes in flushing messages

Create New Transmission History Store

Note:

For additional details on configuring MQTT Transmission, see the documentation [here](#)