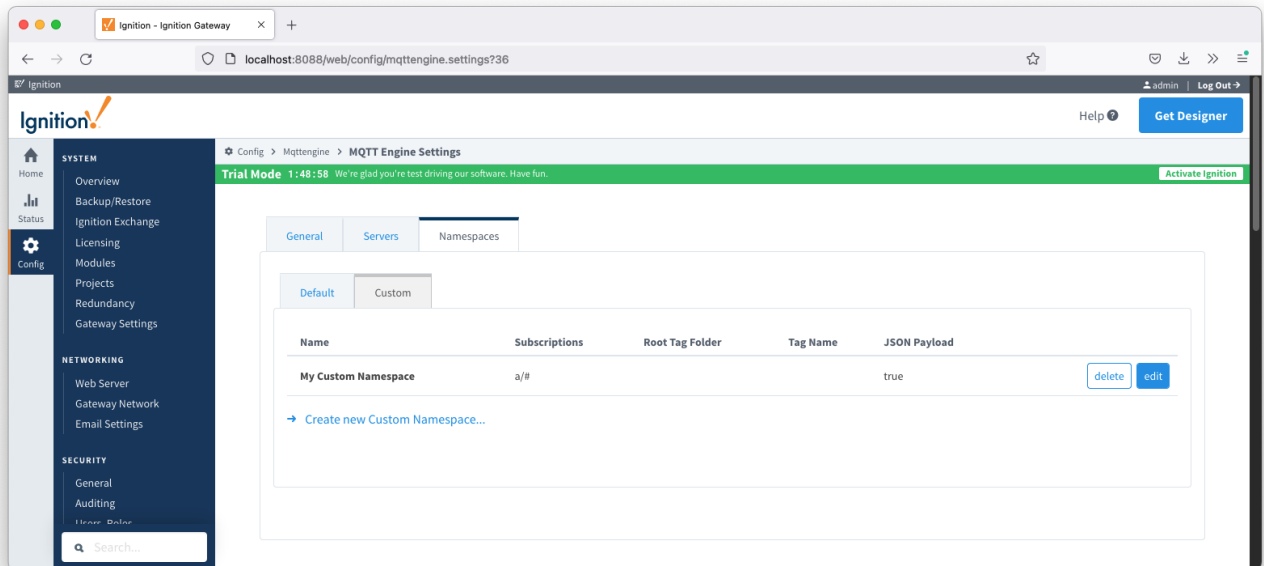# MQTT Engine String Replacement

Prerequisites

- Knowledge of Ignition and Module installation process: Cirrus Link Module Installation
- Have Ignition 8.0.16 or greater installed
- Have MQTT Engine 4.0.10 or later installed

Abstract

MQTT Engine supports 'String Replacement' for both MQTT Topics and Payloads for character sequences that will become part of an Ignition tag path. In many cases with Custom Namespaces this can be required if one does not have full control of the topics and payloads that are sent. This can also be true with certain Sparkplug based payloads where metric names may have unusual characters in them. While MQTT and Sparkplug both support characters such as . & % =, Ignition does not support these as valid characters in a tag path or tag name. As a result, sometimes it may be necessary to tell MQTT Engine to replace certain characters or strings of characters with something else so the tag path and tag names can be properly created in Ignition.

String Replacement Usage

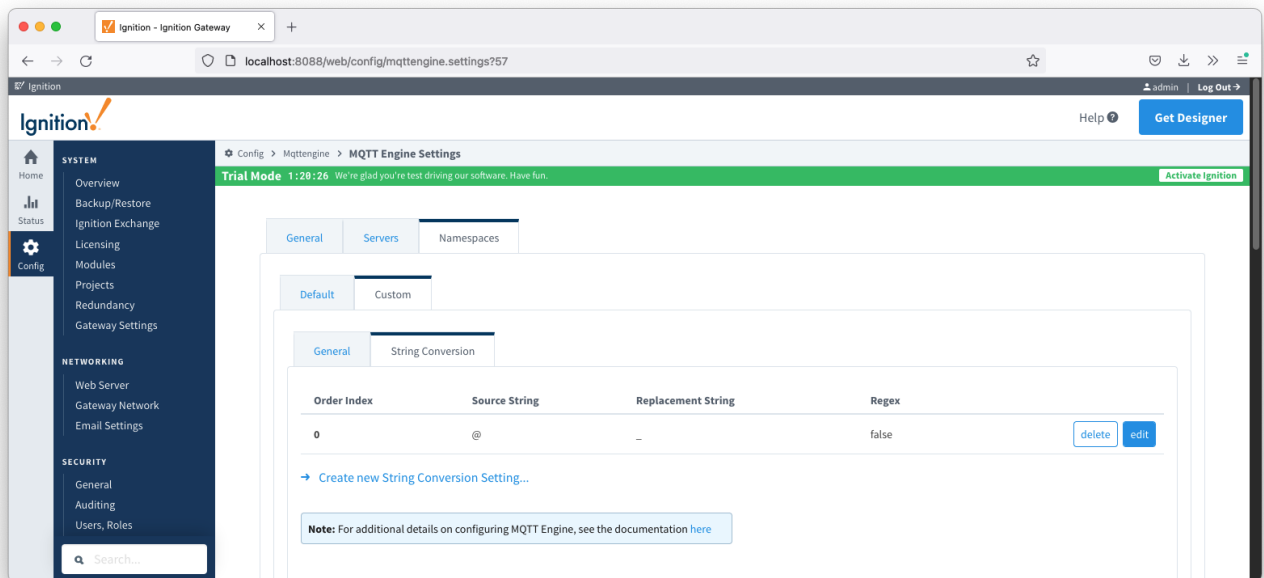Consider the following Custom Namespace.

Now consider an incoming MQTT message with the characteristics:

- Topic = a/b
- Payload = { "tag 1": 123, "tag@2": "some.value" }
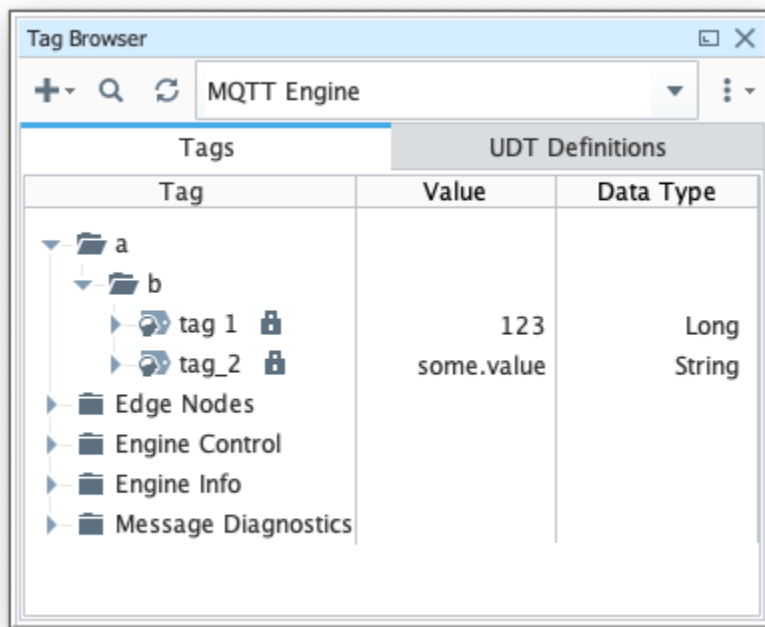
This will fail to be handled by the Custom Namespace and an error will be logged that looks like:

```
INFO   | jvm 1    | 2022/01/06 11:33:29 | E [c.c.m.e.g.j.JsonPayloadHandler] [19:33:29]: Failed to handle
tagPath - not adding tag: a/b/tag@2 - Remove it from future payloads
```

Using 'String Conversions' can allow this tag to be created. Consider the following String Conversion:

This tells MQTT Engine that for this custom namespace, convert any '@' characters that would normally be part of the tag path to '_' characters. So now when we publish the same MQTT message, we don't get an error message and the tag gets properly created as shown below with the '@' character replaced by a '_' character.

The String Conversion feature can be used in any Custom Namespace or in the Sparkplug B Namespace. The String Replacement feature supports the following.

- Replacements are executed using the 'Order Index' order from lowest to highest.
- Single character to single character replacement
- String to string replacement
- Character or string to nothing (i.e. removes a string or character)
- Regular Expression (regex) to string, character, or nothing replacement (i.e. uses a regex for pattern matching rather than a literal string)
- Character replacements will only occur for characters that will be part of the tag path. Character replacements are not applied to Ignition tag values.

Examples:

- Remove all '@' symbols

Order Index	Source String	Replacement String	Regex		
0	@		false	delete	edit

- Replace all '@' symbols with '_'

Order Index	Source String	Replacement String	Regex		
0	@	_	false	delete	edit

- Replace all 'abc' character sequences with 'def'

Order Index	Source String	Replacement String	Regex		
0	abc	def	false	delete	edit

- Replace all characters between the literal strings 'start' and 'end' with a '_' character using a regex

Order Index	Source String	Replacement String	Regex		
0	(?<=start)(.*)(?=end)	_	true	delete	edit

Additional Resources

- Inductive Automation's Ignition download with free trial
  - Current Ignition Release
- Cirrus Link Solutions Modules for Ignition
  - Ignition Strategic Partner Modules
- Questions about this tutorial?
  - Check out the Cirrus Link Forum: https://forum.cirrus-link.com/
  - Contact support: support@cirrus-link.com
- Sales questions
  - Email: sales@cirrus-link.com
  - Phone: +1 (844) 924-7787
- About Cirrus Link
  - https://www.cirrus-link.com/about-us/