Determining the settings for an Injector History Store

⚠ In release 4.0.20, major improvements were made to the Injector disk-backed History Store and upgrading is highly recommended.

Click here for Ignition Version Compatibility

Introduction

Store and Forward allows data to be buffered locally at a client when connections are down to a cloud endpoint and deliver that data when the connection is restored. This feature is critical in most applications because if we lose connection to the cloud end point, we will lose data if it is not buffered locally. When Store and Forward is enabled and the edge node detects a disconnect to the cloud end point, messages will be stored locally in a history store. When the edge node can reconnect to the cloud end point, it will publish any stored messages. The rate at which these stored messages are flushed from the history store is configurable to prevent any delays in the delivery of live data.

Determining the settings for an Injector History Store requires understanding the unique system properties at each Edge Node. There are a number of factors involved in determining how much history data can be stored including but not limited to system resources such as CPU, RAM (especially when using 'In-Memory'), Disk IOPS (if using 'Disk-Backed'), the nominal message rate (e.g. number of messages being pushed per second in the system), the flush rate, bandwidth availability, etc.

Configuring Store and Forward

Store and Forward requires the following parameters:

- **Type**
  - The type of History Store with options of In-Memory and Disk-Backed (available in 4.0.17)
  - Data stored in an In-Memory History Store will not be persisted across a module configuration change, module disable/enable, module restart or power loss.
  - Data stored in a Disk-Backed History Store will persist across a module configuration change, module disable/enable, module restart or power loss.
- **Message Capacity - deprecated in 4.0.19**
  - The maximum number of messages to store before dropping the oldest historical messages
- **History Max Age**
  - Maximum number of minutes to store history before dropping the data.
  - Default 44640 (31 days)
- **Flush Period**
  - The period to wait in milliseconds between publishes when flushing messages upon reestablishing communication.
  - The publishing of historic data messages is interleaved with the publishing of live data messages.
  - Default 100

How to determine these settings

Testing is the best way to determine your settings for your Edge Gateway(s) and the recommended approach is to:

- Restart the Injector module which will clear the Injector Info tags
- Monitor the Injector Info History Store tags for live updates on the count of messages pushed and average size of the messages.
  - Review the following documents to see the Injector Info tags available:
    - AWS Injector Info Tags
    - Azure Injector Info Tags
    - Google Cloud Injector Info Tags
- Extrapolate based on the test to determine the memory requirement for how long you wish to store the data.

Once you have determined the size of the memory store required, you can configure the History Store setting the History Max Age and Flush Period.

- Introduce a network failure and monitor the Messages Stored tag to verify that messages are being stored correctly
- Resolve the network failure and monitor the Messages Pushed tag to verify the messages are being pushed correctly and the rate does not impact the live data publishing