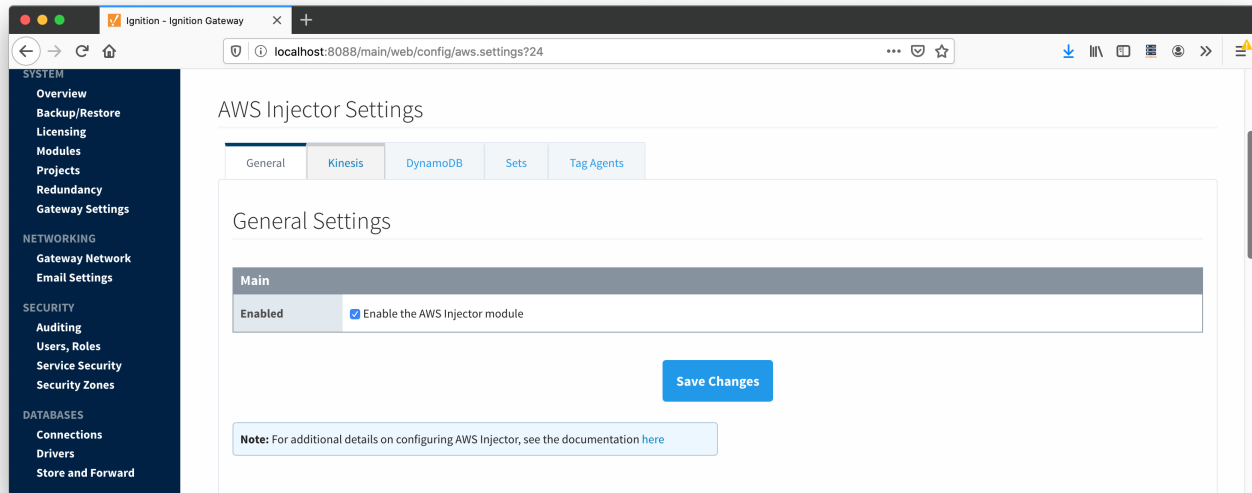


# AWI: Configuration

The AWS Injector module provides the ability to push Tag data to an Kinesis Streams and DynamoDB Database endpoints. The settings configuration for this module are located under the Configure tab of the Ignition Gateway web UI. Once in the AWS Injector Settings configuration section there are five tabs: General, Kinesis, DynamoDB, Sets, and Tag Agents. Each of these tabs is described in detail in the following sections.

## General

The first tab contains the general settings for the AWS Injector module.



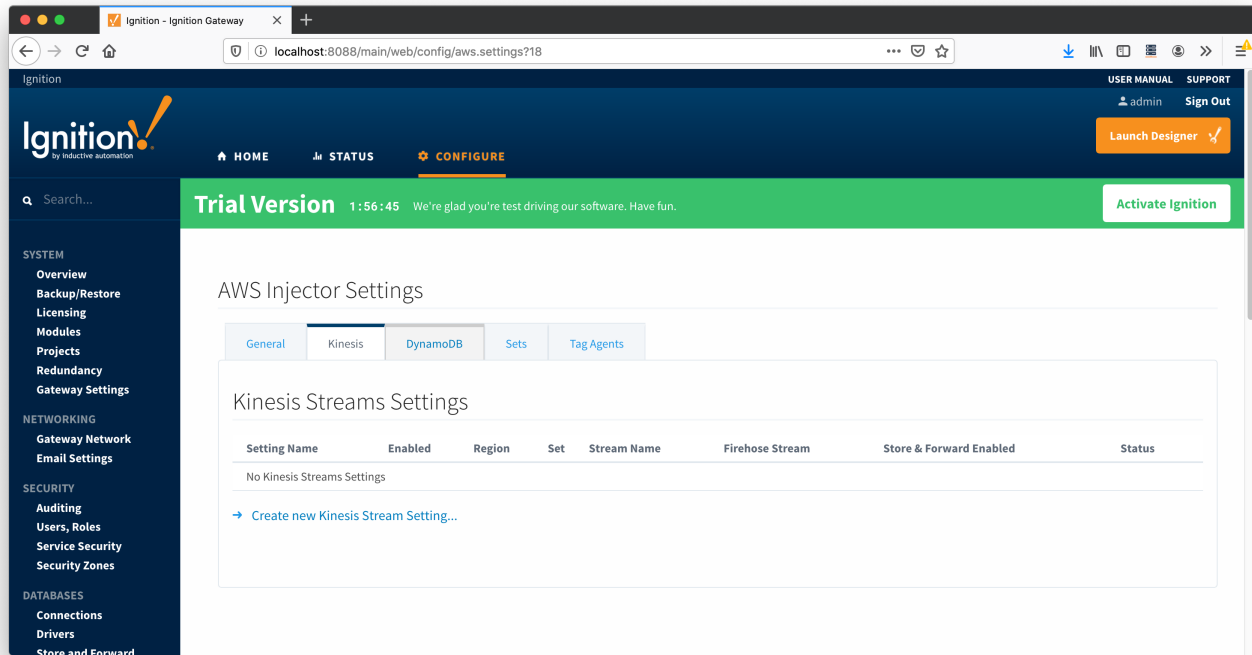
The general configurations options available on this tab are listed below:

### Main

- **Enabled**
  - Sets whether the module is enabled or disabled. If disabled, the Tag Agents will not run and now data will be pushed to any configured endpoints.

## Kinesis Streams

The next tab is the list of AWS Kinesis Stream endpoints that the module should connect to to push Tag data.



One or more AWS Kinesis Stream endpoints can be configured on this tab. The configuration options for an AWS Kinesis Stream connection are listed below.

## Main

- **Setting Name**
  - This is a friendly name of the AWS Kinesis Stream used to easily identify it. This must also be unique.
- **Enabled**
  - Whether or not this stream is enabled to connect to the Kinesis endpoint
- **AWS Access Key**
  - The AWS Access Key ID.
  - Example: AKIAIOSFODNN7EXAMPLE
- **Password/AWS Secret Key**
  - The AWS Secret Access Key.
  - Example: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
- **AWS Session Tokens**
  - Whether to use AWS session tokens
  - **Role ARN**
    - The AWS role ARN
  - **Session Duration**
    - The AWS session duration
  - **Session Name**
    - The AWS session name
- **Region**
  - The AWS region.
  - Example: us-east-1
- **Set**
  - The Set to associate this AWS Kinesis Stream connection with.
- **Stream Name**
  - The name of the AWS Kinesis Stream.
- **Firehose Stream**
  - Enables pushing data directly to Firehose

## Store & Forward

- **Store & Forward Enabled**
  - Whether to enable Store & Forward capabilities for this stream
- **Store & Forward Type**
  - The type of the Store & Forward mechanism
- **Message Capacity**
  - The Maximum number of messages to store before dropping the oldest historical messages
- **Flush Period**
  - The period of time to wait (in milliseconds) between sending when flushing messages

## Advanced

- **Partition Key Count**
  - The number of unique indexed partition keys
- **Partition Key**
  - An optional partition key pattern that supports token substitution for the following message fields: \${group}, \${edgenode}, \${device}. This should only be configured if explicitly defined partition keys are necessary when inserting into the Kinesis Stream.
- **Max Message Size**
  - The maximum message size allowed to be pushed before messages will be broken up into smaller messages (in bytes).

Clicking on the "Create new Kinesis Stream Setting..." link will bring up the following form for adding a new AWS Kinesis Stream endpoint.

The screenshot shows a web browser window with the URL `localhost:8088/main/web/config/aws.settings?23`. The page title is "AWS Injector Settings". On the left is a dark blue sidebar with a navigation menu. The main content area has tabs for "General", "Kinesis", "DynamoDB", "Sets", and "Tag Agents". The "Kinesis" tab is selected, and the form is titled "New Kinesis Stream Setting".

**General** | **Kinesis** | DynamoDB | Sets | Tag Agents

### New Kinesis Stream Setting

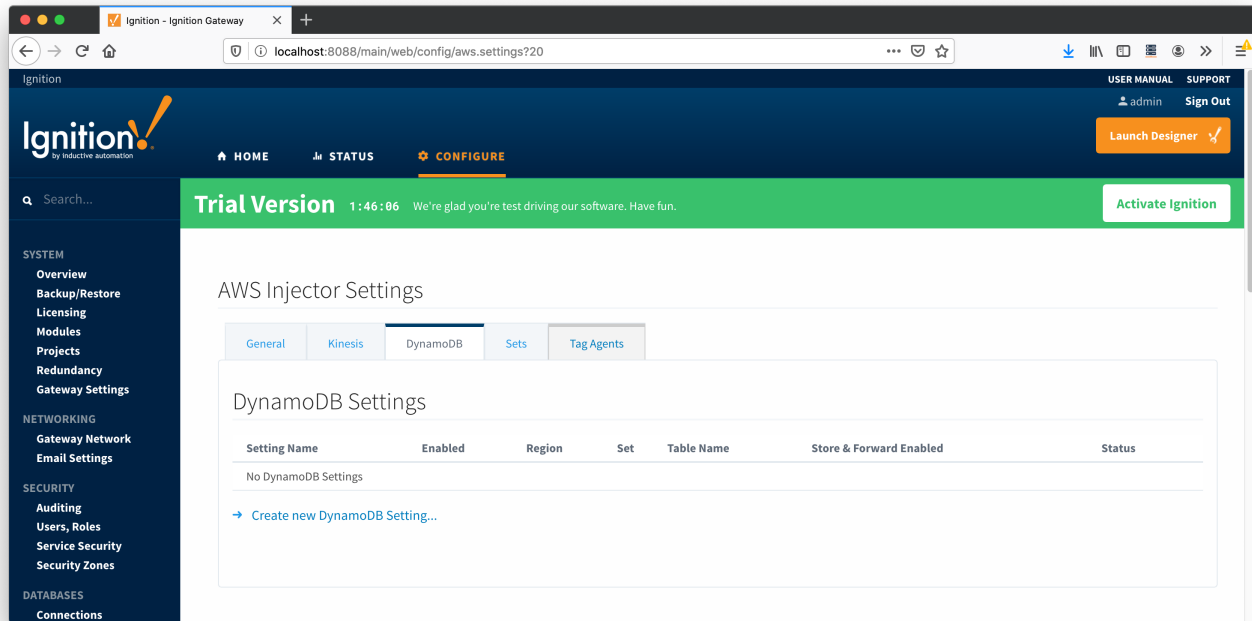
| Main               |                                                                                                    |
|--------------------|----------------------------------------------------------------------------------------------------|
| Setting Name       | <input type="text" value="New Kinesis Stream"/><br>A friendly name for this Kinesis Stream setting |
| Enabled            | <input checked="" type="checkbox"/> Enable this setting                                            |
| AWS Access Key     | <input type="text"/><br>AWS Access Key                                                             |
| Password           | <input type="password"/><br>AWS Secret Key                                                         |
| Password           | <input type="password"/><br>Re-type password for verification.                                     |
| AWS Session Tokens | <input type="checkbox"/> AWS Session Tokens                                                        |
| Role ARN           | <input type="text"/><br>AWS Role ARN                                                               |
| Session Duration   | <input type="text"/><br>AWS Session Duration                                                       |
| Session Name       | <input type="text"/><br>AWS Session Name                                                           |
| Region             | <input type="text" value="us-east-1"/><br>AWS Region                                               |

**Navigation Menu (Left Sidebar):**

- 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

## DynamoDB Databases

The third tab is the list of AWS DynamoDB Database endpoints that the module should connect to to push Tag data.



One or more AWS DynamoDB Database endpoints can be configured on this tab. The configuration options for an AWS DynamoDB Database connection are listed below.

## Main

- **Setting Name**
  - This is a friendly name of the AWS DynamoDB Stream used to easily identify it. This must also be unique.
- **AWS Access Key**
  - The AWS Access Key ID.
  - Example: AKIAIOSFODNN7EXAMPLE
- **Password/AWS Secret Key**
  - The AWS Secret Access Key.
  - Example: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
- **Region**
  - The AWS region.
- **Set**
  - The Set to associate this AWS DynamoDB Database connection with.
- **Table Name**
  - The name of the AWS DynamoDB Database table.

## Store & Forward

- **Store & Forward Enabled**
  - Whether to enable Store & Forward capabilities for this stream
- **Store & Forward Type**
  - The type of the Store & Forward mechanism
- **Message Capacity**
  - The Maximum number of messages to store before dropping the oldest historical messages
- **Flush Period**
  - The period of time to wait (in milliseconds) between sending when flushing messages

Clicking on the "Create new DynamoDB Setting..." link will bring up the following form for adding a new AWS DynamoDB Database endpoint.

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

## AWS Injector Settings

General Kinesis **DynamoDB** Sets Tag Agents

### New DynamoDB Setting

|                |                                                                  |
|----------------|------------------------------------------------------------------|
| Main           |                                                                  |
| Setting Name   | New DynamoDB Stream<br>A friendly name for this DynamoDB setting |
| Enabled        | <input checked="" type="checkbox"/> Enable this setting          |
| AWS Access Key | <input type="text"/><br>AWS Access Key                           |
| Password       | <input type="text"/><br>AWS Secret Key                           |
| Password       | <input type="text"/><br>Re-type password for verification.       |
| Region         | us-east-1<br>AWS Region<br>(default: us-east-1)                  |
| Set            | Default<br>The Set this DynamoDB setting is associated with      |
| Table Name     | <input type="text"/><br>DynamoDB Table Name                      |

|                         |                                                                                |
|-------------------------|--------------------------------------------------------------------------------|
| Store & Forward         |                                                                                |
| Store & Forward Enabled | <input type="checkbox"/> Enable Store and Forward capabilities for this stream |

## Sets

The Sets tab contains a list of AWS Sets. Each set represents a grouping of AWS endpoints. When a set is referenced by a Tag Agent the Agent will push Tag data to all AWS endpoints contained within that Set. The Sets are disjoint, meaning that a single AWS Kinesis Stream or DynamoDB Database endpoint cannot be in more than one set. Out of the box the AWS Injector module will have one "Default" set defined.

## AWS Injector Settings

General Kinesis DynamoDB **Sets** Tag Agents

| Name    | Description | Push Policy |
|---------|-------------|-------------|
| Default | Default set | PUSH_TO_ALL |

[Create new AWS Set...](#)

Additional Sets can be configured for situations where multiple Tag Agents will need to push to different AWS endpoints. The configuration options for Sets are listed below.

## Main

- **Name**
  - This is the friendly name of the set used to easily identify it.
- **Description**
  - This is a friendly description of the set.
- **Push Policy**
  - This defines which endpoints to push to. If PUSH\_TO\_ALL is selected, every endpoint that is part of this set will receive all messages. If PUSH\_TO\_ANY is selected, only one of the endpoints that is part of this set will receive any given message. PUSH\_TO\_ANY is useful when adding endpoint configurations to increase the throughput of the Injector.

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

The screenshot shows the 'AWS Injector Settings' page with the 'Sets' tab selected. The 'New AWS Set' form contains the following fields:

- Name:** A text input field with the value 'New Set'. Below it is the description: 'The friendly name of this Set'.
- Description:** A text input field. Below it is the description: 'Description of this Set'.
- Push Policy:** A dropdown menu with 'PUSH\_TO\_ALL' selected. Below it is the description: 'The Push Policy defines whether all cloud end-points in a set will be pushed to or only one will at a time'.

A blue button labeled 'Create New AWS Set' is located at the bottom right of the form.

## Tag Agents

Tag Agents are the workers within AWS Injector that monitor tag events, convert them to a JSON representation, and push them to one or more AWS endpoints. Out of the box the AWS Injector module will have one "default" Tag Agent defined.

The screenshot shows the 'AWS Injector Settings' page with the 'Tag Agents' tab selected. The 'Agents Settings' section contains a table with the following data:

| Tag Provider Name | Tag Path   | Convert UDTs | Publish UDT Definitions | Set     | Auto-discover Tags | Group 1 |
|-------------------|------------|--------------|-------------------------|---------|--------------------|---------|
| default           | Edge Nodes | true         | true                    | Default | false              |         |

Below the table is a link: [→ Create new Agent Settings...](#)

Tag Agents are configured to point to a single folder. All Tags within that folder will be monitored by the Tag Agent.

## Tag Settings

- **Tag Provider Name**
  - The name of the Tag provider containing the tags.
- **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, in milliseconds, that Tag events will be aggregated into a single payload before pushing.
- **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**
    - This can only be set if 'Convert UDTs' is false
    - Whether or not to push the UDT Definitions in the the NBIRTH messages
- **Set**
  - The Set of AWS endpoints that the Tag Agent will push to.
- **Auto-discover Tags**
  - Whether newly added tags should be dynamically scanned and their values pushed. This field is disabled by default. It should remain disabled while manually editing tags and/or their configurations. It should only typically be enabled in systems where tags are created in real time.

## 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.

The Sparkplug settings are optional and allow for an additional customization of how the Tag Agent scans and discovers tag within the specified Tag Path. Here is a brief description of how the Agent scans/discovers folders based on the different combinations of potential Sparkplug Settings.

- If all three IDs are left blank the Agent will assume the following folder structures follow the Tag Path:
  - `<groupFolder>/<edgeNodeFolder>/<deviceFolder>/<tags>`
  - `<groupFolder>/<edgeNodeFolder>/<tags>`
- If only the Group ID is specified the Agent will assume the following folder structure follows the Tag Path:
  - `<edgeNodeFolder>/<deviceFolder>/<tags>`
  - `<edgeNodeFolder>/<tags>`
- If the Group ID and the Edge Node ID are specified the Agent will assume the following folder structure follows the Tag Path:
  - `<deviceFolder>/<tags>`
  - `<tags>`
- If the Group ID, Edge Node ID, and the Device ID are specified the Agent will assume the following folder structure follows the Tag Path:
  - `<tags>`

As you can see, the Sparkplug settings can be defined by setting these IDs in this configuration page, or leave them blank so that the Agent will scan and discover them based on the tag tree layout.

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

