Ignition MQTT Security Context

Abstract

MQTT Security Context allows secure command writes through MQTT Engine to MQTT Transmission by using custom tag permissions to authorize a tag write based on user.

The primary purpose of using this feature is to use Ignition's internal 'Security Context' objects to validate writes to tags from MQTT Engine to Transmission. Internally, Ignition checks that any user or application has the permissions to perform a write to a tag in Ignition. If the user/application is allowed, Ignition permits this to occur.

However, if a user/application has permissions to write to the MQTT Engine tag, it doesn't necessarily mean that the user has permissions to write to the remote corresponding MQTT Transmission tag. By enabling this feature, we enforce permission validation on the MQTT Transmission side and we do this by sending the Security Context object from MQTT Engine to MQTT Transmission. We can then use that object to validate the user/applications permissions on the MQTT Transmission side.

In order to do this in a secure way, the user security context is encrypted and included with the published write command message from MQTT Engine. At MQTT Transmission, the security context is decrypted and, if the user is authorized to write to the tag, this results in a successful write and the tag change is published. If the user is not authorized to write to the tag, there is no action taken.

Tags must have write permission enabled. Tag write permissions can applied to the entire tag provider or on individual tags to guarantee the write security.

These tag writes can also be configured to create an entry in the Ignition Audit Log

> ⚠ To use the MQTT Security Context feature you must be using MQTT Engine and MQTT Transmission modules 4.0.10 or greater and Ignition 8.1.11 or greater
>
> To use the Audit Log Record feature you must be using MQTT Transmission module 4.0.16 or greater

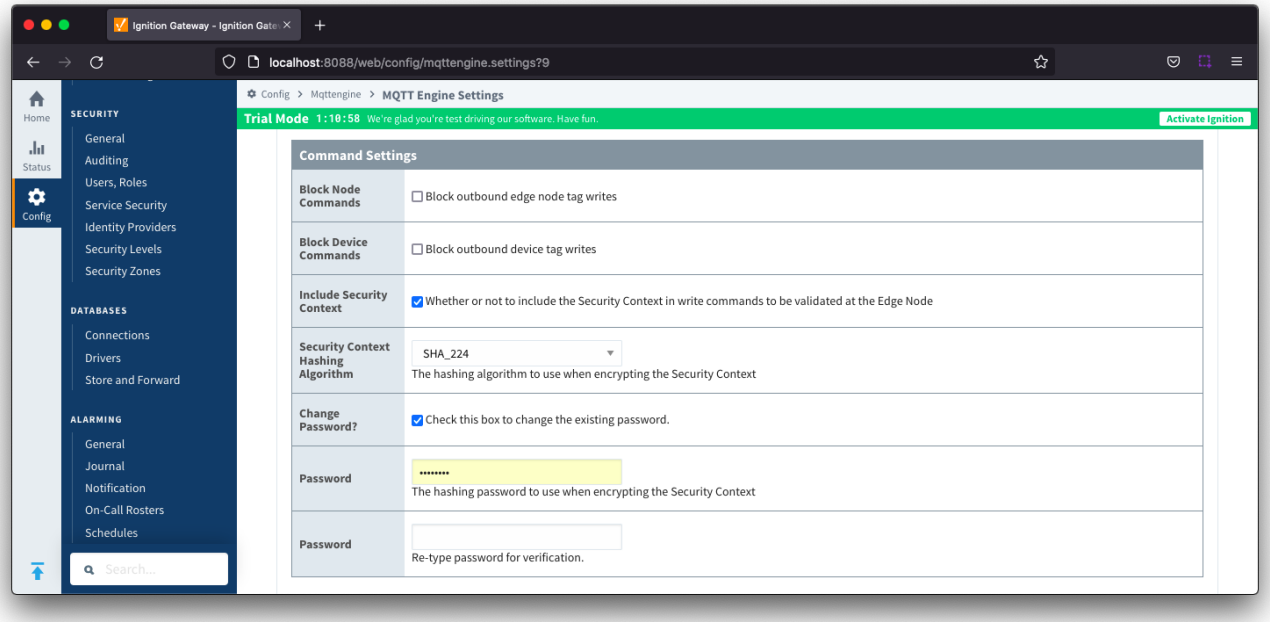
Central Gateway Configuration

MQTT Engine

In the Ignition Gateway web UI, navigate to the MQTT Engine Settings in the left side bar. From the Main tab, set the following elements in the Command Settings section.

- Select checkbox **Include Security Context** in write command to be validated at the Edge Node
- Select the **Security Context Hashing Algorithm** algorithm to use when encrypting the Security Context. Options include SHA_1, SHA_224, SHA_256, SHA_384 and SHA_512
- Select checkbox **Change Password?** and set the **Password** to be used when encrypting the Security Context

> ⚠ Block Node Commands and/or Block Devices Commands must be de-selected for the **Include Security Context** feature to be enabled



Edge Device Configuration

MQTT Transmission

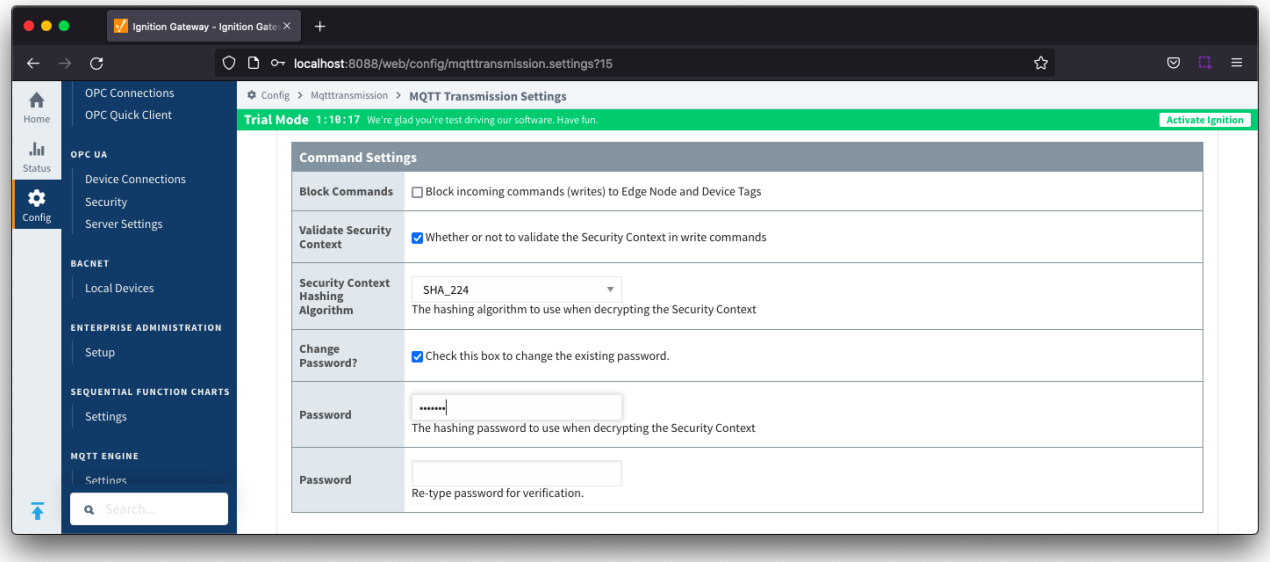
In the Ignition Gateway web UI, navigate to the MQTT Transmission Settings in the left side bar. From the Transmitters tab, for each transmitter set the following elements in the Command Settings section.

- Select checkbox **Validate Security Context** to validate the security context in write command
- Select the **Security Context Hashing Algorithm** algorithm to use when decrypting the Security Context.

Select checkbox **Change Password?** and set the **Password** to be used when encrypting the Security Context

The Security Context Hashing Algorithm and the Password must be the SAME as configured at MQTT Engine.

Block Commands must be de-selected for the **Validate Security Context** feature to be enabled

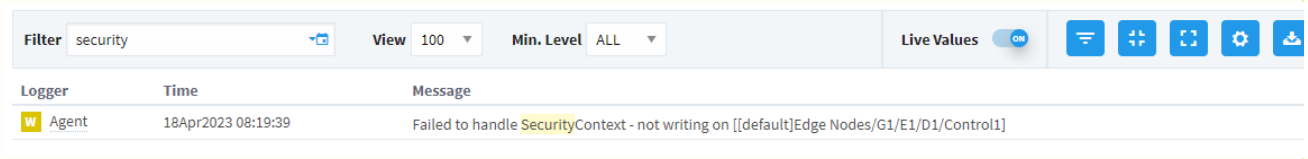


Creating an Audit Log Record

Tag writes using the security context can be configured to create an Ignition audit log record following the steps below:

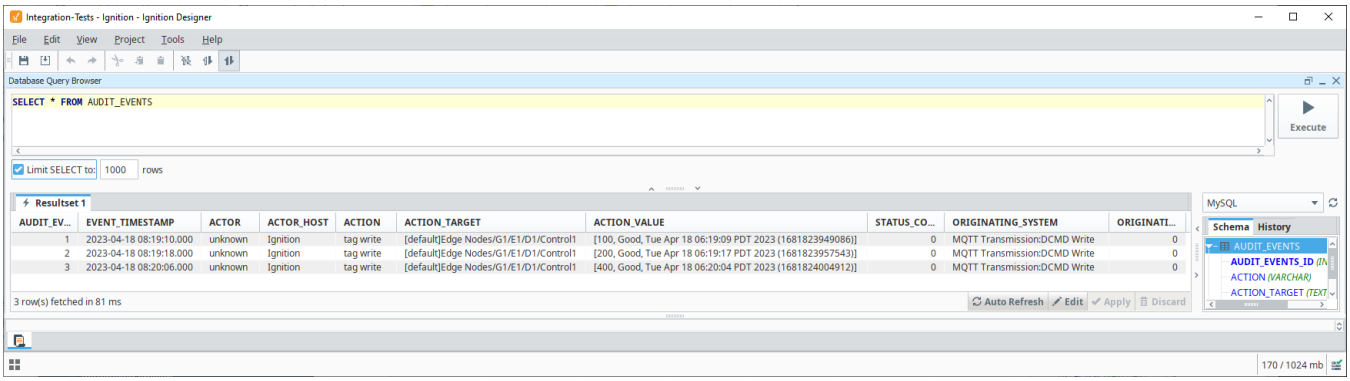
- Create a Database Audit Profile
- On the MQTT Transmission General tab, set the Audit Profile parameter to your database audit profile

The audit log will only include successful tag writes. If a tag write is attempted with an invalid security context, the com.cirruslink.mqtt.common.gateway.agent.Agent will create a Warning log entry similar to the one shown below



Use any of the standard Ignition ways to view the the Audit Log System. The image below shows the records through the Database Query Browser and the fields are populated as:

Name	Description of value
EVENT_TIMESTAMP	The timestamp that the tag write using the security context was performed in the format YYYY-MM-DD HH:MM:SS:mmm
ACTOR	Set to "unknown" (Ignition 8.1.33 or lower) Set to username (Ignition 8.1.34 or higher and MQTT Engine 4.0.20 or higher)
ACTOR_HOST	The originating host system gateway name
ACTION	Set as "tag write"
ACTION_TARGET	The tag path for the tag that is being written to
ACTION_VALUE	The Qualified Value (value, quality, timestamp) for the tag write
STATUS_CODE	Currently not used - set as "0"
ORIGINATING_SYSTEM	The system generating the audit record - set as "MQTT Transmission:DCMD Write"
ORIGINATING_CONTEXT	Currently not used - set as "0"



Additional Resources

- Inductive Automation's Ignition download with free trial
  - Current Ignition Release
- Cirrus Link Solutions Modules for Ignition
  - Ignition Strategic Partner Modules
- Support questions
  - 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/