HARTING

Pushing Performance

People | Power | Partnership

# User Guide

## MQTTs Broker Container

# Table of Contents

1st Edition 2019

# 1. Introduction

## 1.1. About the MICA MQTTs Broker Container

The MQTTs broker container provides an MQTT message broker for the MICA with extensive configuration options including limiting messages to specific topics and support for secure communication using SSL certificates.

This document contains an overview and an explanation of each function provided by the container. It also shows the steps to manage access control and security configuration for the broker.

## 1.2. MICA MQTTs Broker Container Features

The MQTTs broker container lets you:

- Broadcast messages on requested topics.
- Define access control lists for topics.
- Secure connections between subscribers and publishers with SSL certificates.

# 2. Installation, Initial Setup, and Configuration

## 2.1. Default Installation

1. Log into the MICA with admin rights.
2. Click *Install.*
3. Click *Select File* and select the installation archive.
4. Click *Execute* to start the installation.
5. The installer will display a readme file with information about the installation archive.
6. During the installation process, you can name the container, e.g. *MQTTs*.
7. Wait until the installation is finished and click *Close.*

## 2.2. Initial Configuration

The installed container is initially turned off and configured for IPv6 only. Right click on the container tile and click S*tart App* to start the container. The container now listens to and broadcasts messages from the standard 1883 MQTT port.

To subscribe to the MQTT broker, you can user the fully qualified container name, for example mqtt://*mqtt.mica-xxx.domain*:1883/ if the container name is *mqtt*, its IP address or URL, or simply mqtt://*mqtt*:1883/ if you are connecting from the same MICA the container is running on.

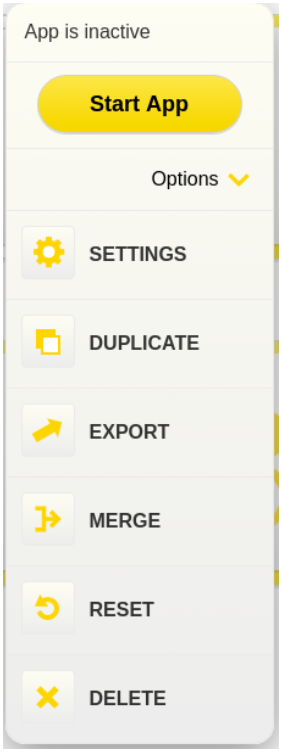
If you need to configure and enable IPv4:

1. Right click on the container click *Options* and then click *Settings*.
2. Expand the IPv4 configuration section.
3. Select the type of configuration which is the best for you. Either *DHCP, Static, Partial DHCP* [1] and enter the required information.
4. Click *Apply* to apply the new settings.

[1] For a more detailed explanation of MICA network configuration, see *MICA Getting Started* available at *harting-mica.com*.

# 3. Basic Container Operations

Right clicking the MQTTs Broker Container tile opens the container context menu. Click Options to access the following functions:




- Start App / Stop App: Starts or stops the container.
- Options: Expands or hides the basic container functions.
- Settings: Shows the container information and lets you configure the IPv4 / IPv6 settings and single sign-on mode.
- Duplicate: Duplicates the container on your MICA.
- Export: Exports the container to your PC or a network drive. All configurations you set will be kept.
- Merge: Overwrites the reset point of the container with its current configuration.
- Reset: Resets the configuration of the MICA MQTTs Broker to the last reset point or the factory default if no merge was executed before.
- Delete: Deletes the container including all its configuration and user data.

# 4. Access Control Lists

By default, all users can publish topics and subscribe to topics on the MQTTs broker. Access control lists let you to define four types of access restriction to topics:

- Read/Write for anonymous access
- Read/Write for authorized users
- Read/Write for a client with a specified ID
- Read/Write for a user with a specified ID

## 4.1. The ACL File Format

The Access Control List file uses the standard Mosquitto acl_file format[2], listing permissions for topics, named users, and all clients successively. For example:

```
# This affects access control for clients with no username
# assuming allow_anonymous is true.

topic read $SYS/#

# This only affects clients with username "roger".

user roger

topic foo/bar

# This allows access for bridge connection messages all clients.

pattern write $SYS/broker/connection/%c/state
```

Topic access is added with lines of the format:

```
topic [read|write|readwrite] <topic>
```

The parameter is optional (unless <topic> includes a space character) - if not given then the access is readwrite.

For more information on Mosquitto configuration file formats, see *https://mosquitto.org/man/mosquitto-conf-5.html.*

## 4.2. Importing Access Control List Files

In the section *Access Control* hover the Access Control File button and click *Import*.


Select an access control list configuration file in the file browser and click Open to import the file and update the configuration.

[2] For more information on Mosquitto configuration file formats, see https://mosquitto.org/man/mosquitto-conf-5.html

The Access Control List file uses the standard Mosquitto acl_file format3. To download a sample access control list file, hover over the Access Control File button and click Export.

## 4.3. Exporting Access Control List Files

In the section *Access Control* hover over the Access Control File button and click *Export*.


If no configuration file has been imported before, the MQTT container will export the default configuration file.

## 4.4. Deleting an Access Control List File

To revert access control list to defaults, In the section *Access Control* hover over the Access Control File button and click *Delete*.


[3] For more information on Mosquitto configuration file formats, see https://mosquitto.org/man/mosquitto-conf-5.html

# 5. Password Lists

You can use password lists to restrict access to MQTT topics.

## 5.1. The Password List Format

Passwords are encoded as simple username:password pairs in separate lines.

```
# Password entry for user roger and password rabbit

roger:rabbit
```

Note that since passwords are stored in plain text, this is not a security feature if users have access to the MICA the MQTT broker is running on.

For more information on Mosquitto configuration file formats, see *https://mosquitto.org/man/mosquitto-conf-5.html.*

## 5.2. Importing Password Files

Select an access control list configuration file in the file browser and click *Open* to import the file and update the configuration.


Enable user verification by disabling the *Allow Anonymous* switch.


The Password file uses the standard Mosquitto format *username:password*. To download a sample password list file, hover over the Password File button and click *Export*.

## 5.3. Deleting a Password File

To delete a password file, hover on the *Password File* button and click *Delete*.

# 6. Configuring Secure Connections

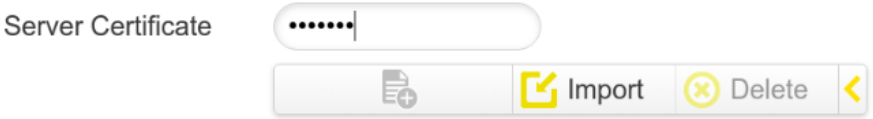
As an additional security layer, you can set up secure connections using SSL certificates.

## 6.1. Adding Certificates

1. Click on the *Security* section title to expand it.
2. Hover over the *CA Certificate* button and click *Import*.


3. Select the CA Certificate in the file browser.
4. Enter the passphrase for the Server certificate file.


5. Hover over the *Server Certificate* button and click and click *Import*.




6. Select the Server Certificate in the file browser.

## 6.2. Deleting Certificates

To delete a certificate, over the certificate button and press *Delete*.


## 6.3. Enabling and Disabling SSL/TLS

Click on the SSL/TLS switch button to enable or disable a secure connection with imported certificates.


## 6.4. Adding Certificate Revocation Lists

Hover over the *Revocation List* button and click *Import* to import a CRL file.


Select the revocation list file in the file browser and click *Import*.

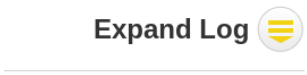
## 6.5. Enabling and Disabling Client Verification

Click on the Client Verification switch button to enable or disable the function.

# 7. Working with Logs

## 7.1. Using the Log Viewer

Click the *Expand Log* to open the log section. Any log entries will be shown in a list.



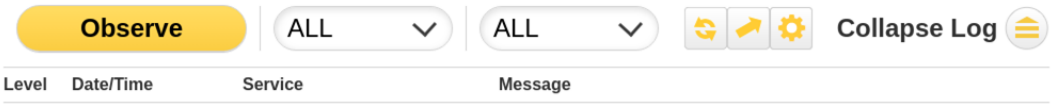

## 7.2. Configuring Log Targets and Log Levels

There are four defined targets:

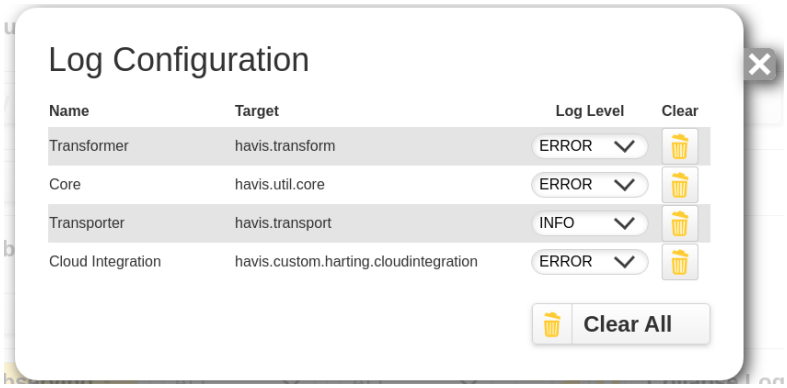
- Transformer – all messages coming from the script transformation module.
- Core – all messages coming from the configuration and base modules.
- Transporter – all messages coming from the communication module.
- MQTT – all messages coming from MQTT broker module.

By default all target levels are set to the ERROR log level apart from the Transporter that is set to the more detailed INFO level.

To configure the log levels, Click Settings in the log section.




In the settings dialog you can set levels for the individual targets.




To clear messages with a specified log level for a target, click the trash icon or the *Clear All* button.

To close the dialog click .

To display the logs live, click *Observe*. It will change the status to *Observing*.


You can filter logs with two combo boxes available next to the observe button. Filters can only be changed when the observing mode is disabled.


1. Disable observing mode if enabled.
2. In the first combo box select a log level of displayed logs.
3. In the second combo box select a target to display.
4. Click *Observe* to start observing logs with enabled filtering.

To see all content of message, click on the log entry with that message.

# 8. REST API

**GET/accesscontrol**

Returns AccessControl object

**DELETE/accesscontrol/acfile**

Delete access control file

**GET/accesscontrol/acfile**

Returns AccessControl file

**HEAD/accesscontrol/acfile**

Returns if AccessControl file exists

**PUT/accesscontrol/acfile**

Upload AccessControl file

**DELETE/accesscontrol/passwordfile**

Delete Password file

**GET/accesscontrol/passwordfile**

Returns Password file

**HEAD/accesscontrol/passwordfile**

Returns if Password file exists

**PUT/accesscontrol/passwordfile**

Upload password file

**GET/security**

Returns Security object

**DELETE/security/trust**

Delete CA Certificate file

**HEAD/security/trust**

Return if CA Certificate file exists

**PUT/security/trust**

Upload CA Certificate file

**DELETE/security/keystore**

Delete Server Certificate file

**HEAD/security/keystore**

Returns if Server Certificate file exists

**PUT/security/keystore**

Upload Server certificate file

**DELETE/security/revoclist**

Delete Revocation List file

**HEAD/security/revoclist**

Returns if Revocation List file exists

**PUT/security/revoclist**

Upload Revocation List file

**PUT/security/passphrase**

Set passphrase