1.0.0

Operating instructions manual

netFIELD App MQTT to OPC UA Converter

DOC220303OI1.0.0EN | Revision 1.0.0 | English | Released | Public

empowering communication

Table of Contents

1 About this document

1.1 Description of the contents

This document describes the **netFIELD App MQTT to OPC UA Converter** from Hilscher. The converter will also be referred to as "MQTT to OPC UA Converter" or "the converter".

Here you will find instructions on how to deploy and configure the application container.

1.2 List of revisions

Index	Date	Author	Revision
1.0.0	2024-06-12	MKE	Document created

Table 1. List of revisions

1.3 Conventions in this document

The terms *Edge Device* and *Edge Gateway* in this document refer to all devices and virtual machines running the *netFIELD Operating System (netFIELD OS)*, including the *netFIELD OS Datacenter* on virtualization platforms.

Admonitions

Notes, operation instructions and results of operation steps are marked as follows:

TIP This is a tip with additional information.

NOTE This is a note that explains something about the text.

IMPORTANT This is a more important version of a note.

CAUTION This is less severe than a warning but more important than a note. Not adhering to this could lead to undesired effects.

WARNING This is a warning. Not adhering to this could lead to data loss or other more severe effects.

User instructions

User instructions in documents will look like this:

→ Goal

a. Step 1
b. Step 2
c. Step n

⇒ Result

2 Introduction

2.1 Brief description

Key features

The **netFIELD App MQTT to OPC UA Converter** "converts" data topics from an MQTT broker (like e.g. *mosquitto*) into OPC UA nodes and provides these nodes to connected OPC UA Clients. The app contains an integrated MQTT client that is capable of subscribing to topics at any available MQTT broker. The user can select and subscribe to individual data topics, which are then forwarded by the app to the Address Space of the integrated OPC UA Server and made available as `string` data types.

Like all netFIELD application containers, the MQTT to OPC UA Converter runs in the **IoT Edge Docker** of a netFIELD Edge Gateway (or netFIELD OS Datacenter) and can be deployed via the netFIELD Portal or on any OCI compliant container engine. The app can be configured either via Remote Control from the netFIELD Portal or via local Ethernet access (see section Configuration for more information).

2.2 General Requirements

Requirements

- netFIELD Edge Gateway or netFIELD OS for virtual machines
- OPC UA Client(s) (on the same device or reachable network) for monitoring the resulting nodes

2.2.1 Memory

The RAM needed by the container depends on the amount of data that you intend to capture and process in your application. We recommend you to provide at least **1 GByte of free RAM** on your Edge Device (ideally 2 GByte or more). In use cases with a stable amount of little data processing, less than 1 GByte RAM might be sufficient.

2.2.2 MQTT broker

The *netFIELD App MQTT to OPC UA Converter* container publishes or retrieves the acquired data via MQTT and thus requires an MQTT broker for operation. The MQTT broker can run on the same host or on a different host machine in your local IT network.

2.2.3 OPC UA Server

The *netFIELD App MQTT to OPC UA Converter* container publishes or retrieves the acquired data as OPC UA nodes and thus requires an OPC UA server to function. The OPC UA Server can run on the same host or on a different host machine in your local IT network.

NOTE | The *netFIELD App MQTT to OPC UA Converter* already contains an OPC UA server.

2.2.4 Limitations

- The embedded OPC UA Server does not support OPC UA write operations
- The embedded OPC UA Server does not support the UA HTTPS transport protocol.
- The embedded MQTT client does not support publishing operations.

3 Use cases

In the use case depicted below, the netFIELD App MQTT to OPC UA Converter runs in the **IoT Edge Docker** of the **netFIELD OS** of an **OnPremise** Edge Gateway. The gateway is connected to an Ethernet network that contains an MQTT Broker. The embedded MQTT client of the MQTT to OPC UA Converter app subscribes to selected topics, converts them into OPC UA nodes and makes them available at its embedded MQTT to OPC UA Converter:

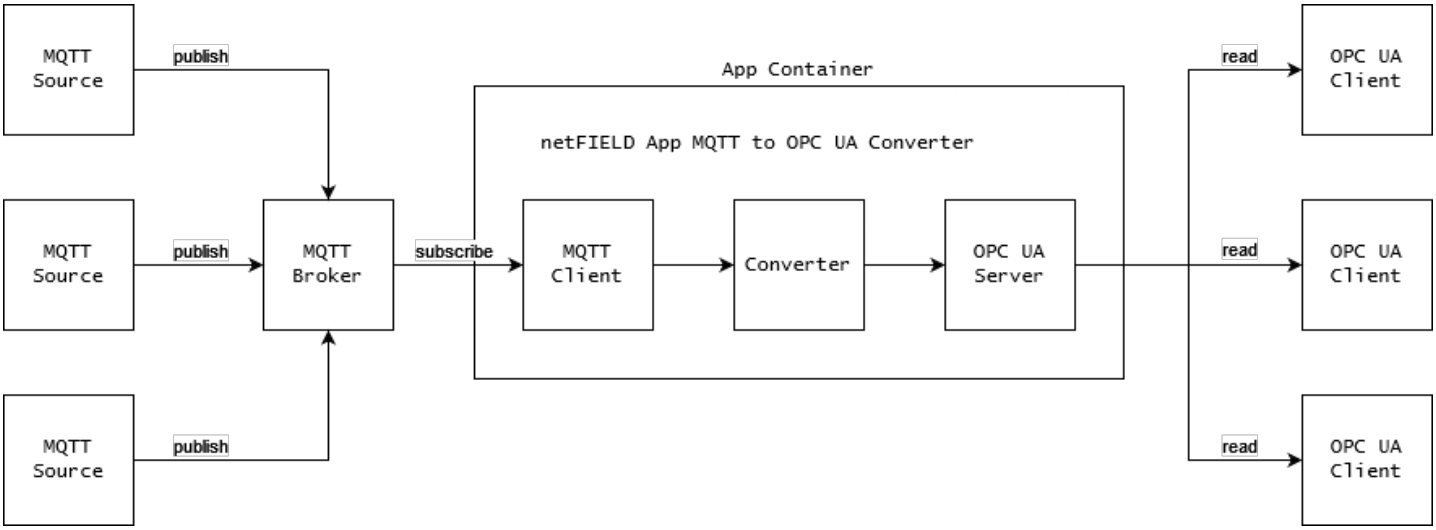


Figure 1. netFIELD App MQTT to OPC UA Converter with MQTT to OPC UA Converter example

Note that the MQTT to OPC UA Converter and/or the MQTT broker could also be running in a different Docker instance or even on a different device, if connected to the OnPremise device via Ethernet.

4 Start parameters of the container

PortBindings

`"4840/tcp"`

Port for the OPC UA Server inside the MQTT to OPC UA Converter.

Binds

Bind	Description
`netfield-app-mqtt-to-opc-ua-converter-data`	Volume for Application data
`/etc/gateway/settings.json`	Optional file which contains the gateway prefix on our systems, the application will use the default value if this is not available. **Snippet 1. Format** `{ "schemaVersion": number, // always 1 "gatewayPrefix": string // default: hardware-ID }`
`/etc/gateway/mqtt-config.json`	Optional file which contains the default MQTT config, the application will create a default config and you can change it via the UI. **Snippet 2. Format** `{ "schemaVersion": 1, "connectTimeout": 300, "serverURIs": [ "tcp://localhost:1883", "tcp://mosquitto:1883" ], "mqttVersion": 3 }`
`/usr/share/ca-certificates/`	CA certificates from the host system.
`/etc/shadow`	User definition and hashed password from the host system.
`/etc/hostname`	Hostname of the host system. Used in the application certificate and endpoint.
`/usr/local`	Path where the UI is copied on our systems and this path is also used to store the PKI of the application.

Table 2. Binds

Environment variables

Variable	Description
`LOGLEVEL`	Specify which severity you want to see in the logs. Possible values: - fatal - error - warn (Default) - info - debug - verbose
`RUNTIME_TARGET`	Specifies if the operation mode of the container. Possible values: - datacenter - netfield - iotedge
`MAXSTRINGLENGTH`	Specifies the maximum length of string which can be forwarded (MQTT payload size). Default: `102400`.

Table 3. Environment variables

5 Configuration

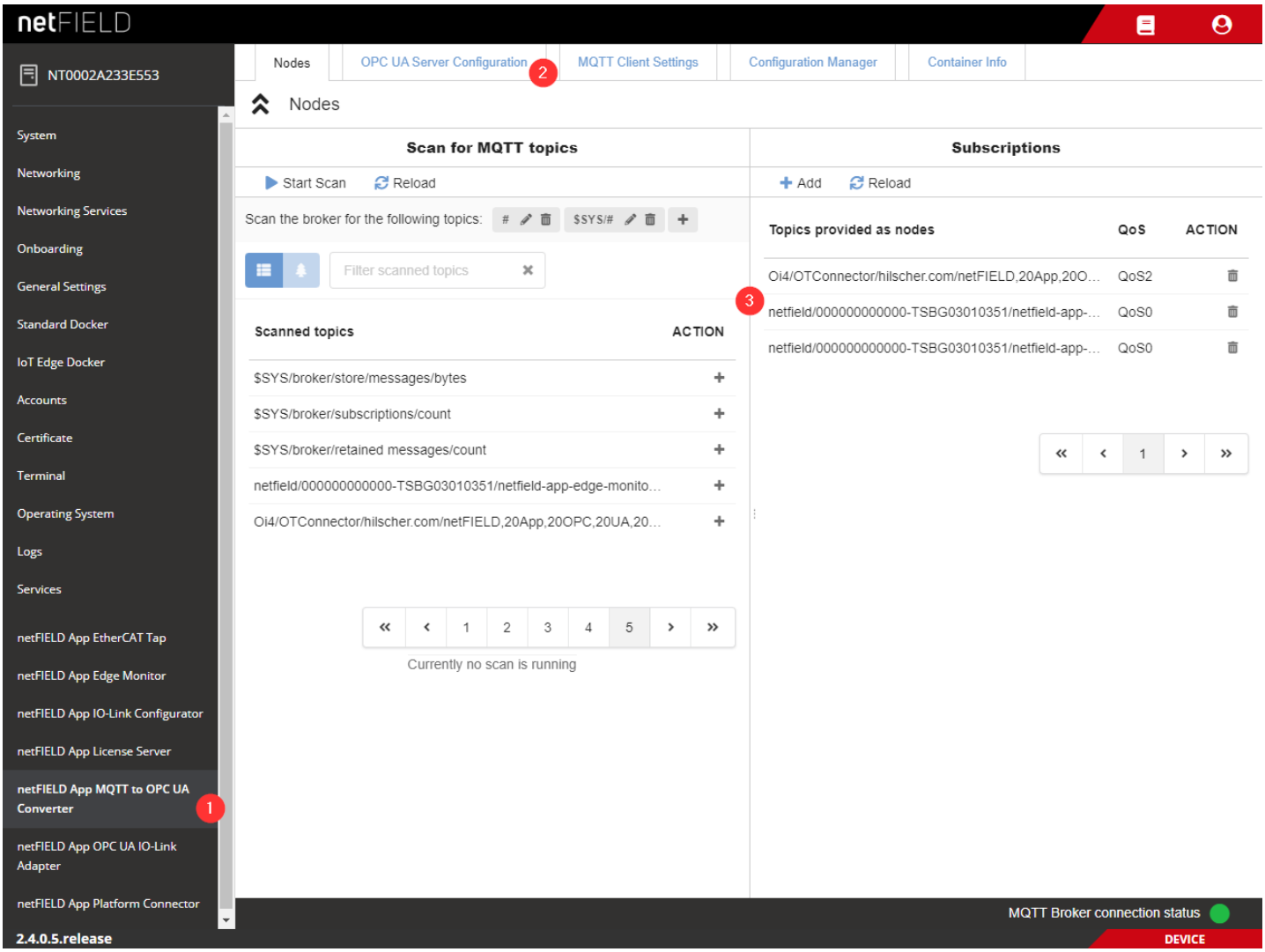
Once the container is deployed, you need to configure which MQTT topics will be converted and configure the integrated OPC UA Server.

Optionally you can configure the MQTT Broker if you do not wish to use the default settings.

5.1 Overview

The netFIELD App MQTT to OPC UA Converter container provides a configuration GUI in the Local Device Manager of the netFIELD OS. This configuration GUI is automatically plugged-in when the container is deployed. After having established a connection to the Local Device Manager (e.g. by Remote Control from the netFIELD Portal, see section *Remote Control* in the *netFIELD Portal* manual, DOC190701OIxxEN), the configuration GUI can be selected in the navigation panel **(1)** of the Local Device Manager.

NOTE Note that it might take a few minutes after deployment before the netFIELD App MQTT to OPC UA Converter entry becomes visible in the navigation panel. You may also have to reload the web page in your browser by pressing F5 on your keyboard.



The tabs in the header of the screen **(2)** allow you to navigate through the configuration and management options of the netFIELD App MQTT to OPC UA Converter. The main screen **(3)** displays the configuration options according to the selected tab.

5.2 Nodes (MQTT topics to OPC UA nodes)

In the **Nodes** tab you can subscribe to the MQTT topics that you want to provide as OPC UA nodes at the embedded MQTT to OPC UA Converter. To see which topics are available, you can "scan" the connected MQTT broker before subscribing to individual topics. As an alternative, you can also directly subscribe to topics if you already know the exact topic string. You can also use wildcards here to subscribe to multiple topics at once.

NOTE You can check your broker connection in the **MQTT Broker connection status** indicator in the footer of the screen. Hover over the indicator to see which MQTT broker you are currently connected to.

After subscribing to a topic, the app immediately provides it as an OPC UA node in its server's address space, allowing external OPC UA clients to read and monitor it. Note that the OPC UA node is a 1-to-1 forwarded `string` type version of the MQTT topic and cannot be edited or renamed.

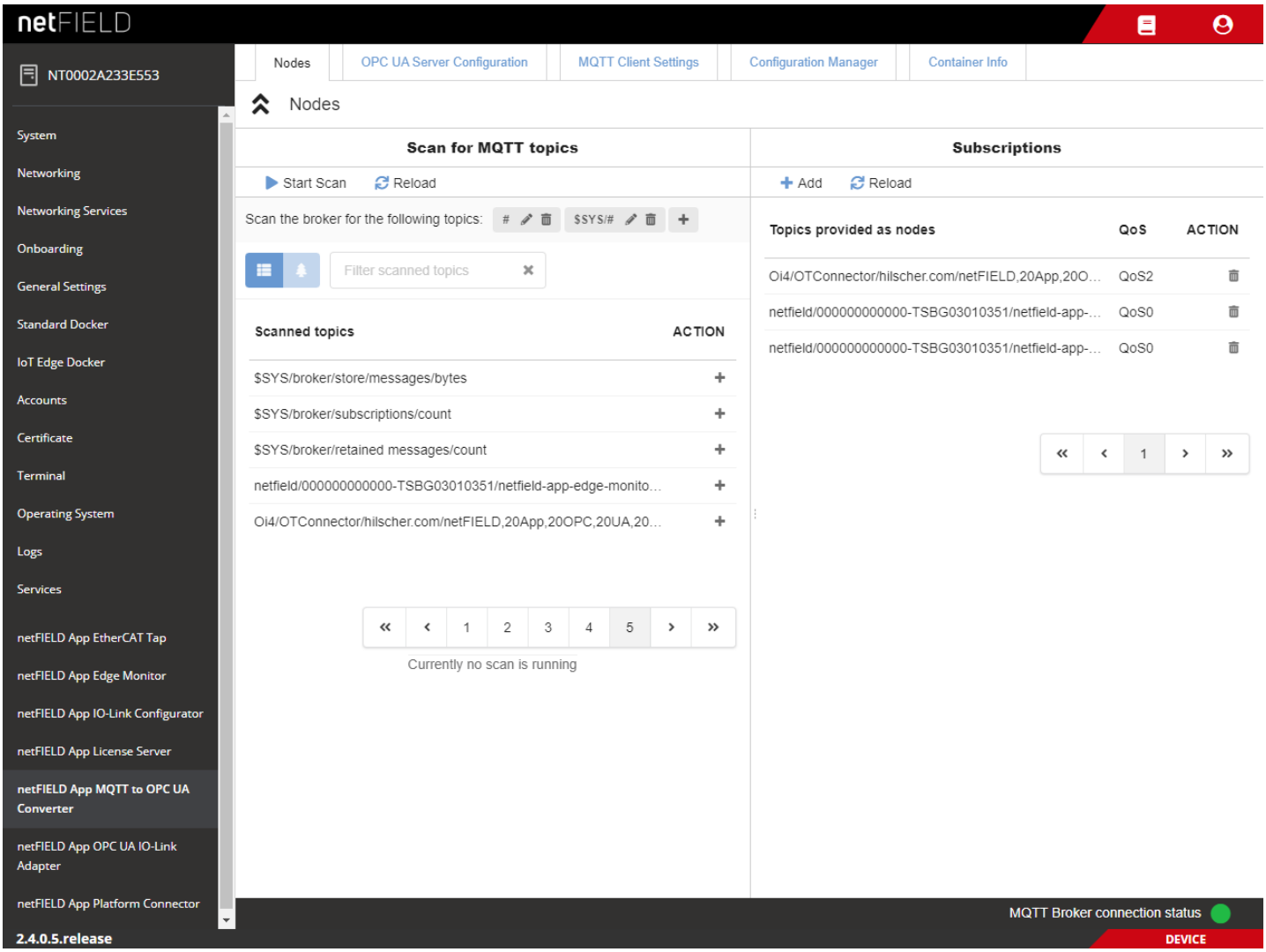
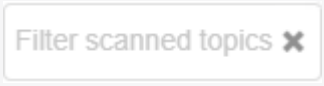


Figure 2. Nodes tab

Element		Description
Scan for MQTT topics	Start Scan	Click here to scan the connected MQTT broker for topics. Before scanning, the **Duration** dialog opens, in which you must specify for how long the embedded MQTT client shall scan the broker. **Note**: The app can detect a topic on the broker only if a message is actually being published to the topic during scan time. It is therefore essential to choose a scan duration that is long enough to catch at least one published message for each topic that you want to retrieve. A running scan is indicated in the footer, together with the estimated end of the scan, e.g.: `"Scan in progress. (est. end at 7/4/2022, 1:14:31 PM)"`
	Reload	Manually refreshes the display of the retrieved MQTT topics. (By default, scan results are automatically refreshed every 30 seconds.)
	Scan the broker for the following topics	Here you can define search conditions for browsing the topics of the MQTT broker. You can explicitly specify a topic (i.e. use the full topic string to retrieve a certain topic) or use the following two wildcard expressions for retrieving your desired topic(s):
		`/+` Single-level wildcard. **Example**: `netfield-app-mqtt-to-opc-ua-converter/+/`
		`/#` Multi-level wildcard. Retrieves all topics matching the string before the wildcard. Must be the final character and preceded by a slash (/#) if used in a topic string. **Example**: `netfield-app-mqtt-to-opc-ua-converter/#` **Note**: If you use # as a stand-alone wildcard (without a preceding topic string), you must omit the slash.
		Note that you must specify at least one search condition. Note also that logical OR operators are applied when you specify multiple conditions. The conditions are preset by default to ensure that all published topics (`#`) and also the broker's system status information topics (`$SYS/#`) are retrieved. To narrow down your search, you can delete the two default conditions and add more specific search conditions.
		Edits the search condition.
		Deletes the search condition.
		Adds a new search condition.
		Changes the display of the retrieved topics from hierarchical ("tree") view to list view and vice versa.
		Fuzzy text search for topics in the scan result list/tree. Acts also as a filter to the displayed topics. To remove the filter, you have to delete all text in the field (e.g. by clicking the **[x]** icon).
	Scanned topics	Displays the scan results.
	ACTION	Opens the subscription dialog (if you are in list view).
		Opens the subscription dialog (if you are in hierarchical view).
		If the list view contains more than ten entries, you can scroll here to display further entries.

Element		Description
Subscriptions	Add	Opens an empty dialog for direct topic subscription. Here you can paste or enter your topic string and/or wildcards without having to "scan" for topics first.
	Reload	Refreshes the display of the subscribed MQTT topics/nodes.
	Topics provided as nodes	Shows a list of the topics to which you have subscribed and which will be provided as nodes in the MQTT to OPC UA Converter.
	QoS	Shows the "Quality of Service" defined for the topic/node.
	ACTION	Delete topic – Deletes the subscription. **Note**: Due to technical reasons, deleted topic subscriptions will still be displayed as nodes in the address space of the MQTT to OPC UA Converter, however without updated message data. To remove these "empty" nodes from the MQTT to OPC UA Converter, you have to restart the container.

Table 4. Elements in Nodes tab

Scanning for topics

a. Make sure that at least one search condition is specified.

> **NOTE** The preset default condition `#` retrieves all topics that are being published to the broker during scan time, while the preset default condition `$SYS/#` retrieves all topics relating to the broker's system status information. To narrow down your search, you can delete the default conditions and add more specific search conditions.

b. Click **[Start Scan]** button.

⇒ The Duration dialog opens:

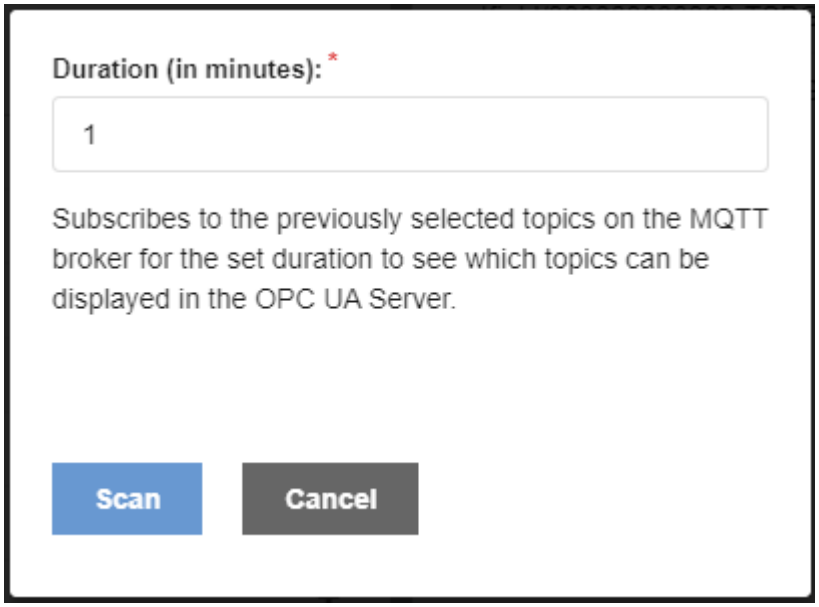


Figure 3. Scan duration dialog

c. In the **Duration** dialog, enter the scan duration time in minutes.

> **NOTE** The app can detect a topic on the MQTT broker only if a message is actually being published to the topic during scan time. It is therefore essential to choose a scan duration that is long enough to catch at least one published message for each topic that you want to retrieve. For example, a scan duration of one minute might not be sufficient to catch a topic that is being published to at an interval of e.g. every two minutes. You would have to choose a duration of at least two minutes to be sure to catch such a two-minute message in order to retrieve the corresponding topic.

d. Click **[Scan]** to start the scan process.

⇒ The app immediately starts to scan the broker for messages that are being published to it by other MQTT clients and lists the thus detected topics under **Topics**. Newly detected topics are added to the list every 30 seconds (or on clicking the Refresh button).

Subscribing to topic(s)

a. To subscribe to a topic (and thus add it as OPC UA node), click the + button next to a topic in the **ACTION** column; or, if in hierarchical view, click the ⊞ button next to the topic level.

⇒ The **Subscribe** dialog opens:

Figure 4. Subscribe dialog

b. If necessary, you can edit the string and e.g. add single level (`/+`) or a multi-level (`/#`) wildcards. Note that the multi-level wildcard `/#` must be placed at the end of the string (i.e. as suffix).

c. In the **QoS** drop-down list, select the MQTT Quality of Service (default is QoS0):

- QoS0: At most once
- QoS1: At least once
- QoS2: Exactly once

> **NOTE** This is the QoS of the message delivery from the MQTT broker to the subscribing client, i.e. to the netFIELD App MQTT to OPC UA Converter. If you define here a QoS that is higher than the QoS defined by the publishing client, the MQTT broker transmits the message with the lower QoS of the publishing client. If you define here a QoS that is lower than the QoS defined by the publishing client, the MQTT broker transmits the message with the lower QoS of the subscribing client (i.e. this app).

d. If you know the exact topic(s) that you want to subscribe to in advance, you don't need to scan the broker first for obtaining a list from which to choose your topic) you can click the + **[Add]** button under **Subscriptions** (right side of the screen) and open an empty subscription dialog in which you can enter or paste your topic string.

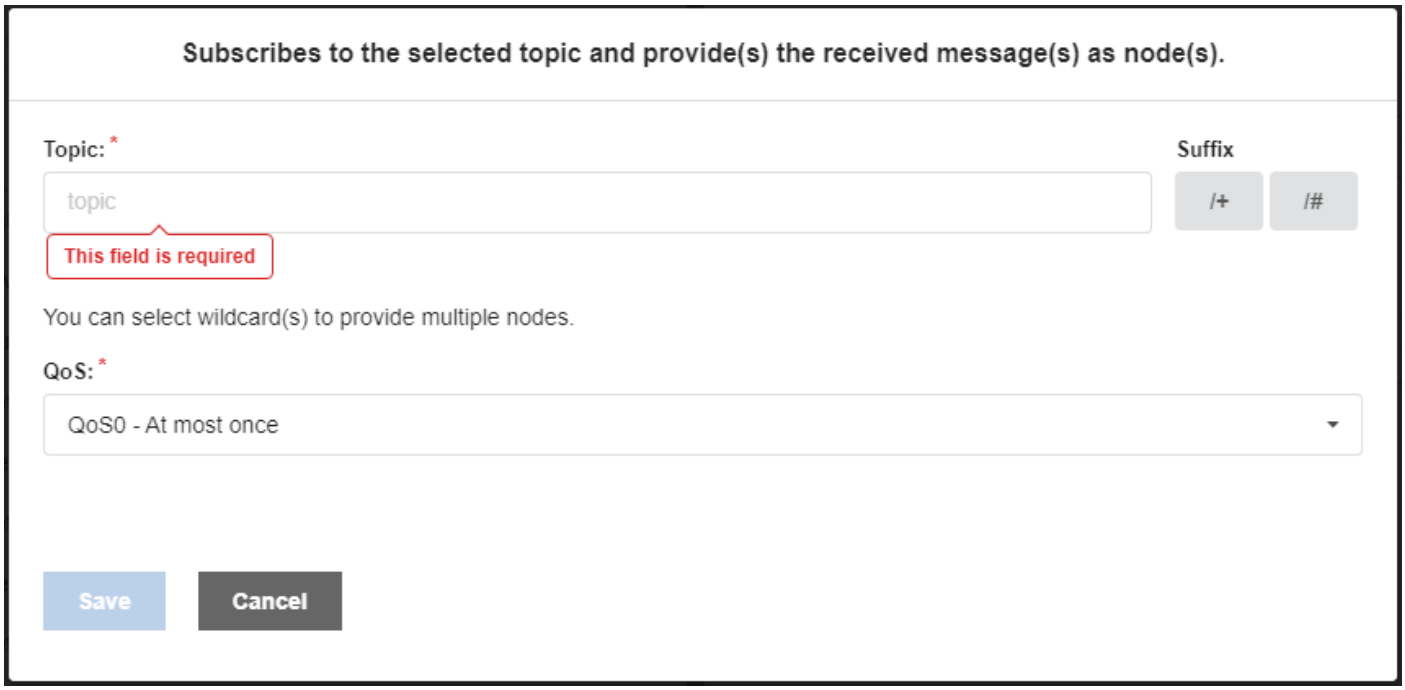


Figure 5. Subscription dialog

⇒ After clicking the **[Save]** button, the app immediately subscribes to the specified MQTT topic(s) and forwards it/them as OPC UA node(s) to the address space of its embedded MQTT to OPC UA Converter, allowing connected OPC UA clients to read and monitor the nodes. The subscribed topics are listed in the **Subscriptions** area on the right side of the screen, where they can also be deleted again, if necessary.

NOTE For technical reasons, deleted topic subscriptions will still be displayed as nodes in the address space of the MQTT to OPC UA Converter; however without receiving and displaying "fresh" data. To remove these "empty" nodes from the MQTT to OPC UA Converter, you have to restart the container. You can restart the container on the IoT Edge Docker page of the Local Device Manager under **Containers › netfield-app-mqtt-to-opc-ua-converter › Restart**.

5.3 OPC UA Server Configuration

In the **OPC UA Server Configuration** tab, you can configure the embedded OPC UA Server, most notably its security settings. Note that OPC UA clients that do not support these security settings will not be able to connect to the server. This page also shows the default Endpoint URL of the embedded OPC UA Server, which you can copy to your clipboard and then paste it into your OPC UA Client application for establishing a connection.

CAUTION By default, the security mode **None** is enabled, which allows accessing the server without the protection of signing and/or encryption. For OPC UA data security in your production environment, we strongly recommend you to disable the **None** option and allow only the **Sign** and/or the **Sign and Encrypt** modes.

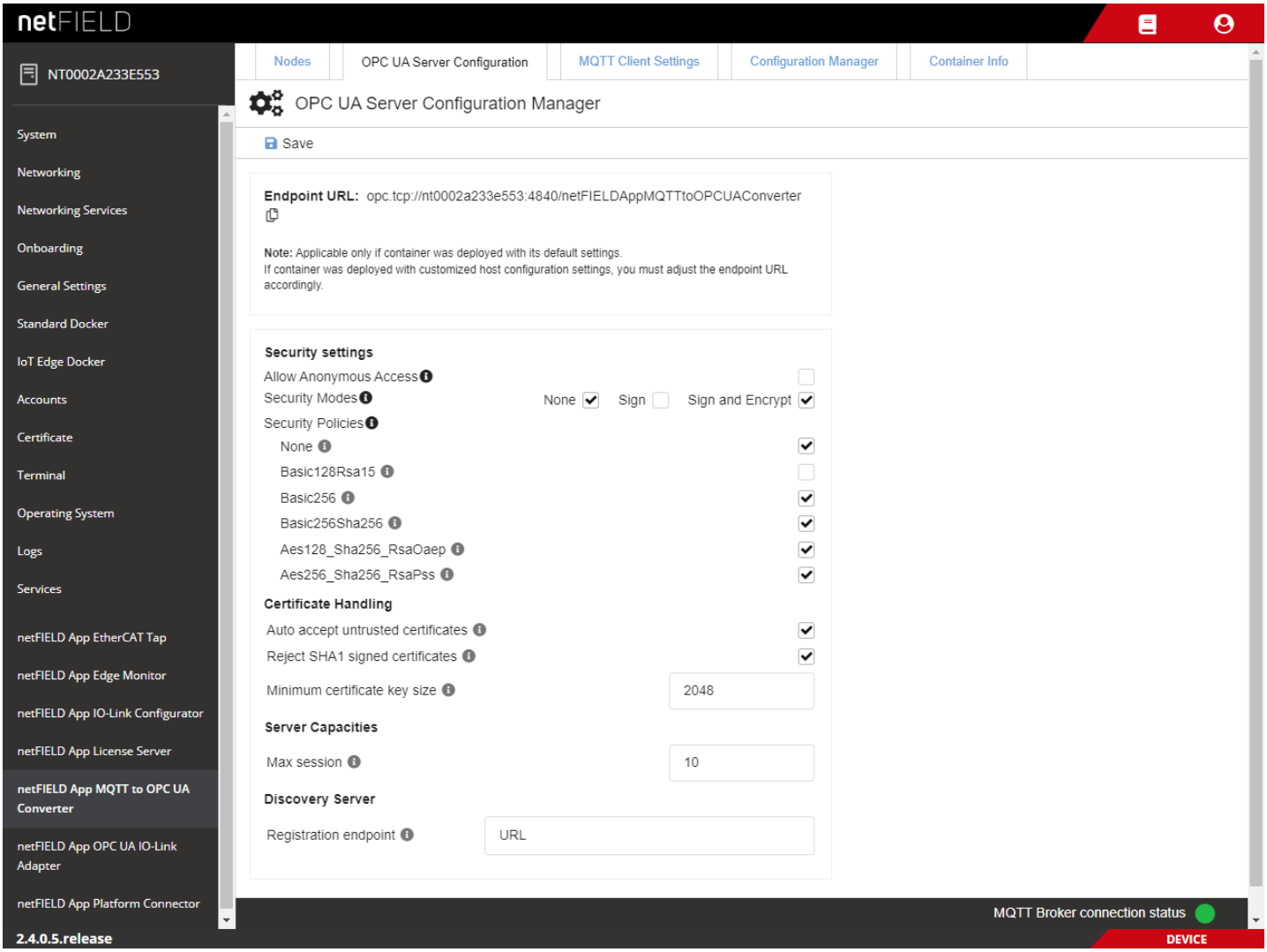


Figure 6. OPC UA Server Configuration Manager

Note the following about **authentication** at the OPC UA Server via username and password: By its default "container create options", the app is bound to the `/etc/shadow` file of Linux and thus uses the user accounts of the netFIELD OS (e.g. the `admin` user) for authentication.

The user accounts of the netFIELD OS can be created and managed on the **Accounts** page of the **Local Device Manager**. Any user defined there can be used for authentication at the OPC UA Server – an assignment of special user roles (like e.g. "Container Administrator") is not required for this. Note also that adding a new user account added via the Local Device Manager is not automatically loaded by the container; you have to restart the container to use a newly created/changed user account for authentication at the OPC UA Server. (You can restart the container on the IoT Edge Docker page of the Local Device Manager under **Containers › netfield-app-mqtt-to-opc-ua-converter › Restart**.)

Settings	Description
Save	Saves your configuration settings after changing.

Endpoint URLs	The pre-configured standard URL of the OPC UA Server is:
	`opc.tcp://[host name]:4840/netFIELDAppOPCUAServer`
	Specify this URL in your OPC UA Client application in order to connect to the server.
	(Click to copy the URL to your clipboard).
	Alternatively, you can use:
	`opc.tcp://[IP address]:4840/netFIELDAppOPCUAServer`
	Note: If you have configured different `HostConfig` settings in the **Container Create Options** (see section [Create Options]) instead of using the container's default settings, you must adapt the Endpoint URL in your client application accordingly.

Security settings	
Allow Anonymous Access	Select this option to allow OPC UA Clients to access the server in "anonymous mode", i.e. without username and password. If this option is disabled, the client has to provide username and password for authentication at the server. Note that the app uses the given accounts of the netFIELD OS for authentication (e.g. the admin user). **Important**: You should not allow "anonymous access" in production environments.

Security Modes	Select here the security modes that the server shall accept. If multiple options are enabled, it is up to the client to choose one of the enabled modes before it establishes a connection:
	`None`
	No protection of data integrity and confidentiality: Data traffic is neither signed nor encrypted. Does not require any security policy or PKI infrastructure. Important: Use only in non-production environments.
	`Sign`
	Ensures data integrity: Signing allows client and server to validate message data, which ensures that data cannot be modified during transfer. This mode requires a PKI infrastructure and the selection of a suitable security policy/algorithm (see below).
	`Sign and Encrypt`
	Ensures data integrity and confidentiality: Data traffic is signed and encrypted, and can thus be neither modified nor read in plain text if intercepted. This mode requires a PKI infrastructure and the selection of a suitable security policy/algorithm (see below).

Settings	Description

Security Policies	Select here the security policies (algorithms) that the server shall support. All policies provide signing, encryption and certificate validation (requires PKI infrastructure). If multiple policies are enabled, it is up to the client to choose one of the enabled policies before it establishes a connection.
	`None`
	Allows connections without security policy.
	Important: **Do not use** in production environments.
	`Basic128Rsa15`
	128-Bit encryption algorithms using RSA15 as a key-wrap algorithm.
	Important: **Do not use** in production environments. This policy supports the SHA1 hash algorithm, which is not considered secure anymore.
	`Basic256`
	256-Bit encryption supporting SHA1 and SHA256 hash algorithms.
	Important: **Do not use** in production environments. This policy supports the SHA1 hash algorithm, which is not considered secure anymore.
	`Basic256Sha256`
	256-Bit encryption supporting SHA256 or stronger hash algorithms. Suitable for high security needs.
	`Aes128_Sha256_RsaOaep`
	Fast and suitable for average security needs.
	`Aes256_Sha256_RsaPss`
	Suitable for high security needs, most secure policy currently available.

Certificate Handling	`Auto accept untrusted certificates`
	Allows the acceptance of "untrusted" certificates, e.g. on platforms without PKI infrastructure.
	`Reject SHA1 signed certificates`
	Automatically rejects certificates signed with the SHA1 hash algorithm (which is not considered secure anymore).
	`Minimum certificate key size`
	Specify the minimum size in bytes required for an auto-generated certificate key (default is `2048` bytes).

Server Capacities > Max session	Specify the maximum number of current sessions allowed by the server (default is `10`). If the limit is reached, new client connections will be rejected.
Discovery Server > Registration endpoint	If you want to register the OPC UA Server at a Local Discovery Server (LDS), you can enter here the URL of the LDS (optional).

Table 5. Settings in OPC UA Server Configuration tab

5.4 MQTT Client Settings

In the **MQTT Client Settings** tab, you can configure the settings of the embedded MQTT client of the MQTT to OPC UA Converter app. By default, the app uses the standard MQTT client settings of the netFIELD OS. According to these default settings, the app first tries to connect to any MQTT broker under the URI `tcp://localhost:1883`, then (if connecting to the first URI fails) tries to connect to a *mosquitto* broker under the URI `tcp://mosquitto:1883`.

If you cannot use these standard MQTT client settings for your MQTT to OPC UA Converter app – because you want to connect it to a different broker (under a different URI and/or with different parameters/credentials) – you must uncheck the **Use general settings** option and enter your new MQTT settings in the configuration fields that are now displayed:

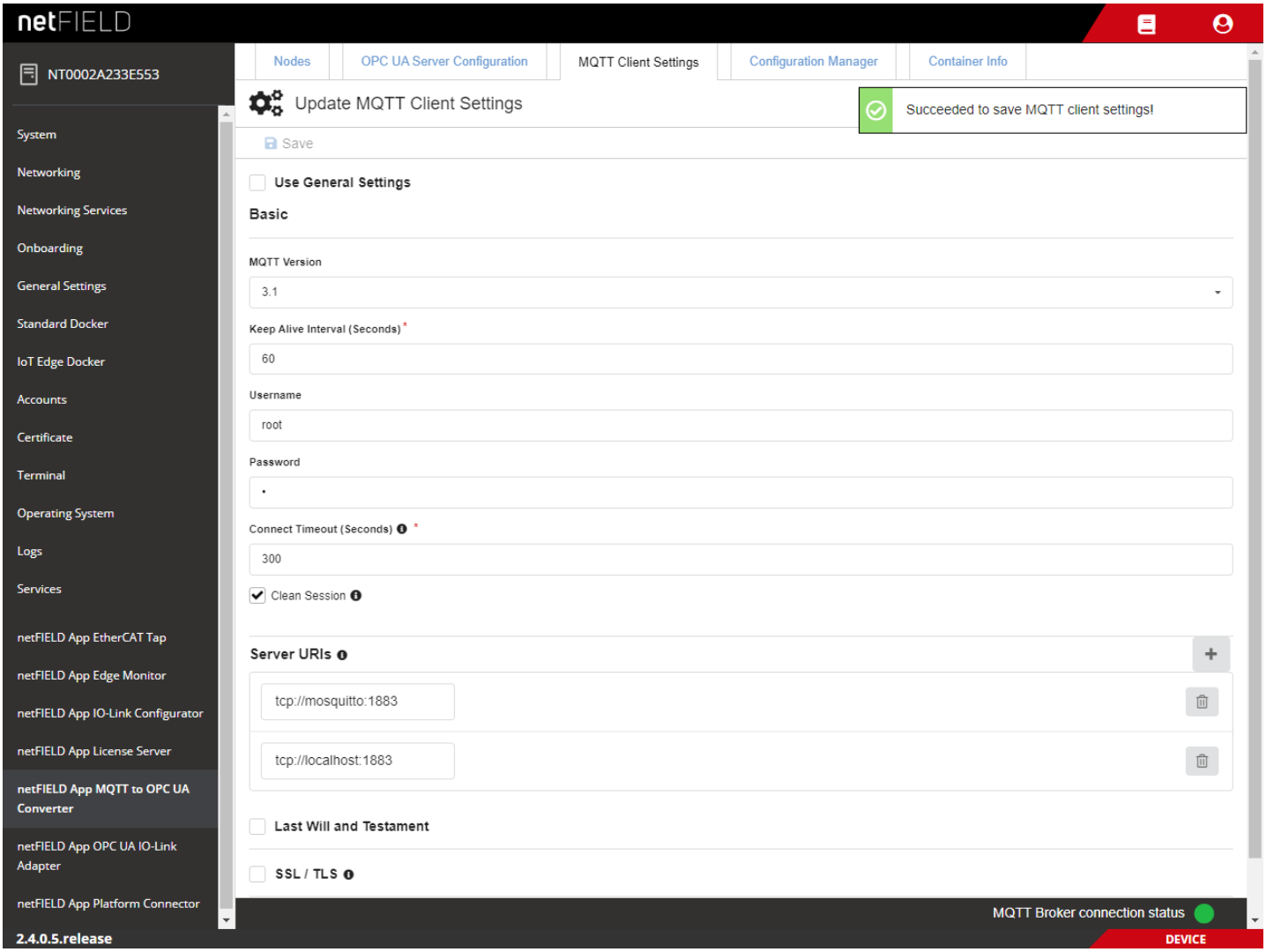


Figure 7. MQTT Client Settings

NOTE Changes to the MQTT Client Settings that you make here for your MQTT to OPC UA Converter app will not affect the standard "global" MQTT Settings of your netFIELD OS.

The standard "global" MQTT client settings of the netFIELD OS can be viewed (and changed if necessary) in the Local Device Manager under **General Settings › Default MQTT Client Settings**.

Element	Description
MQTT version	MQTT version to be used (depending on the MQTT broker).
Keep alive interval (Seconds)	Defines the maximum length of time in seconds that the broker and client may not communicate with each other.
User name	User name for authentication at the broker (if implemented and required by the broker). Note that the *mosquitto* broker deployed from the netFIELD Portal does not require login authentication.
Password	Password for authentication at the broker (if implemented and required by the broker). Note that the *mosquitto* broker deployed from the netFIELD Portal does not require login authentication.
Connect timeout (Seconds)	Defines the maximum length of time in seconds that is allowed for completing the connection process.
Clean session	If **Clean session** is selected, the client does not want a persistent session (meaning that if the client disconnects for any reason, all information and messages that are queued from a previous persistent session are lost). If **Clean session** is unchecked, the broker creates a persistent session for the client.
Server URIs	Server URI of the MQTT broker. **Note**: When multiple server URIs are specified, the client will try to connect to each server one after the other, starting with the first server in the list. If a server connection is successfully established, only this connection will be used. The client will not open multiple connections to multiple servers simultaneously.

Last Will and Testament	Select this option if you want to use the "last will and testament" (LWT) feature of MQTT. (I.e. to notify other clients about an unexpected loss of connection to the broker)
	
	`Topic name`
	Topic name of LWT message
	
	`Retained`
	LWT will be retained on Broker even if a client has already received it.
	
	`Quality of Service`
	QoS of LWT message
	
	`Message`
	Message text, e.g. "unexpected loss of connection"

SSL / TLS	Select this option if you want to use SSL/TLS encryption for creating a secure connection to the MQTT broker.
	
	`File name and path to private key in PEM format`
	Path to the private key on the device; e.g.: `/etc/ssl/private/client-key.pem`
	
	`File name and path to certificate chains in PEM format`
	Path to the certificate chains on the device; e.g.: `/etc/ssl/services/client-cert.pem`
	
	`Override the trusted CA certificates in PEM format`
	Path to override the trusted CA certificates on the device; e.g.: `/etc/ssl/services/ca-cert.pem`
	
	`Enable verification of the server certificate`
	If this option is disabled, the MQTT to OPC UA Converter app will also accept invalid certificates from the broker (not recommended).
	
	Note: This option is for expert users only! In the standard use case, in which the *mosquitto* broker and the OPC UA Converter app are running on the same device, a secure SSL/TLS connection is not necessary (because the connection is "internal" and the overhead of the secure connection can thus be avoided). If you want to use SSL/TLS encryption anyway, see section Using SSL/TLS encryption (optional) for further information.

Table 6. MQTT Client Settings

a. Click Save button to save your new MQTT Client Settings.

⇒ The **Succeeded to save MQTT client settings** message appears.

b. Check the **MQTT Broker connection status indicator** in the footer to see if the connection to the new server has been successfully established:

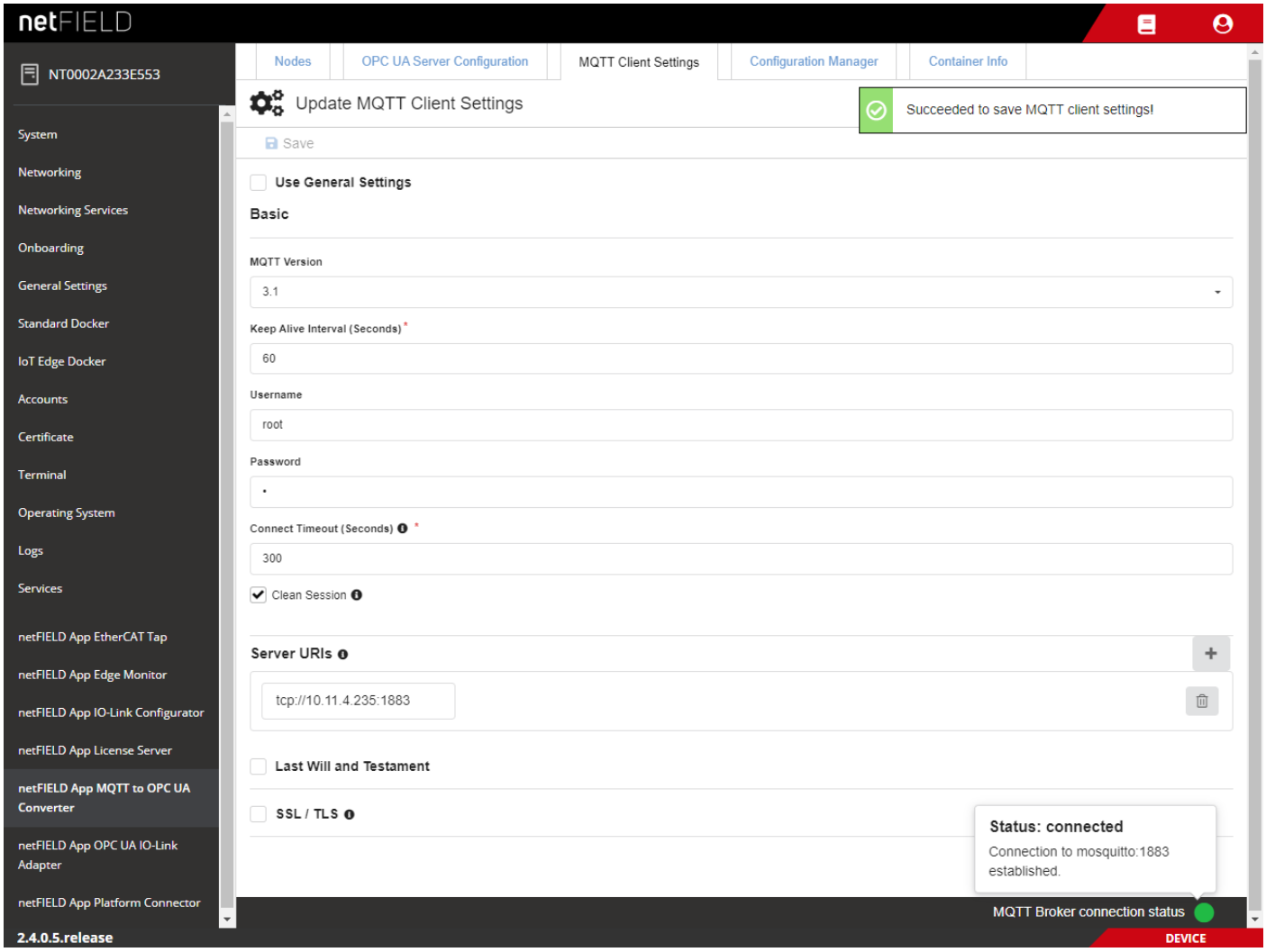


Figure 8. MQTT server connection status indicator in footer

5.5 Configuration Manager

In the **Configuration Manager** tab, you can save the MQTT to OPC UA Converter app configuration settings to your local PC. You can also restore a formerly saved configuration by uploading the configuration file. The download/upload function allows you to practically "clone" your configuration and use it in other MQTT to OPC UA Converter instances (e.g. running on other netFIELD Edge Devices or netFIELD OS Datacenters).

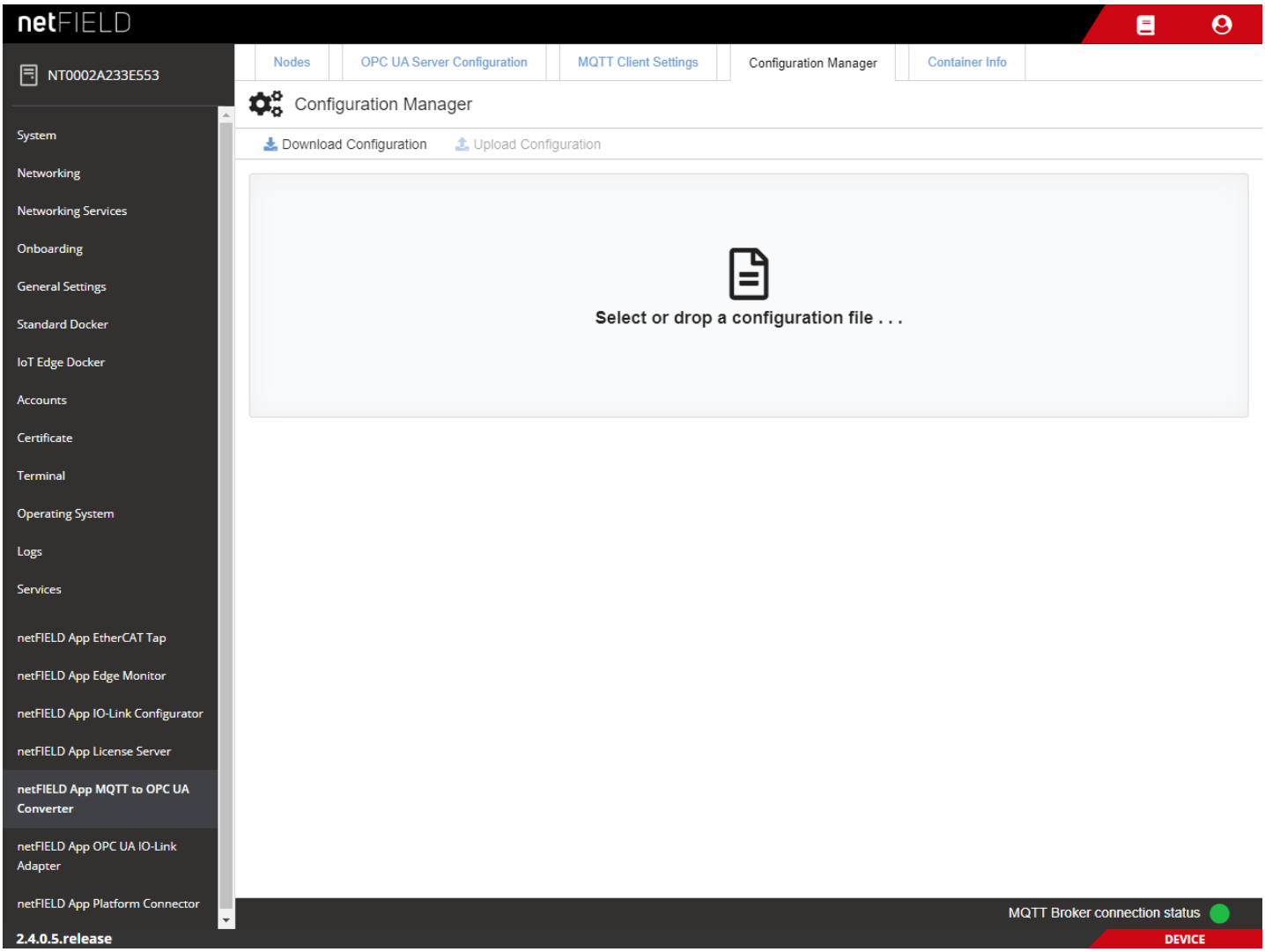


Figure 9. Configuration Backup

Save configuration

1. To save your current configuration, click **[Download Configuration]** button.

   ⇒ The configuration settings are saved to your local PC as ZIP file. The name of the ZIP file is made up by the gateway prefix, app name and date/time of the download. The download path depends on the settings of your web browser.

   NOTE The "gateway prefix" is by default the device ID of the Edge Device on which the MQTT to OPC UA Converter app is deployed. If you are using the app as "standalone" app - i.e. if the app has not been deployed in the IoT Edge Docker via netFIELD Portal -, the "gateway prefix" is the host name of the device.

Restore/import configuration

To restore a formerly saved configuration (or import it into other instances), you must first select the configuration ZIP file by dragging and dropping it from your desktop onto the grey field (as an alternative, you can open the standard Windows file selection dialog by clicking into the grey field).

After having selected the file, the **[Upload Configuration]** button is enabled, and you can now "load" the configuration by clicking the button.

IMPORTANT The Upload Configuration function will overwrite the current configuration settings. We recommend you to save your current configuration before using this function.

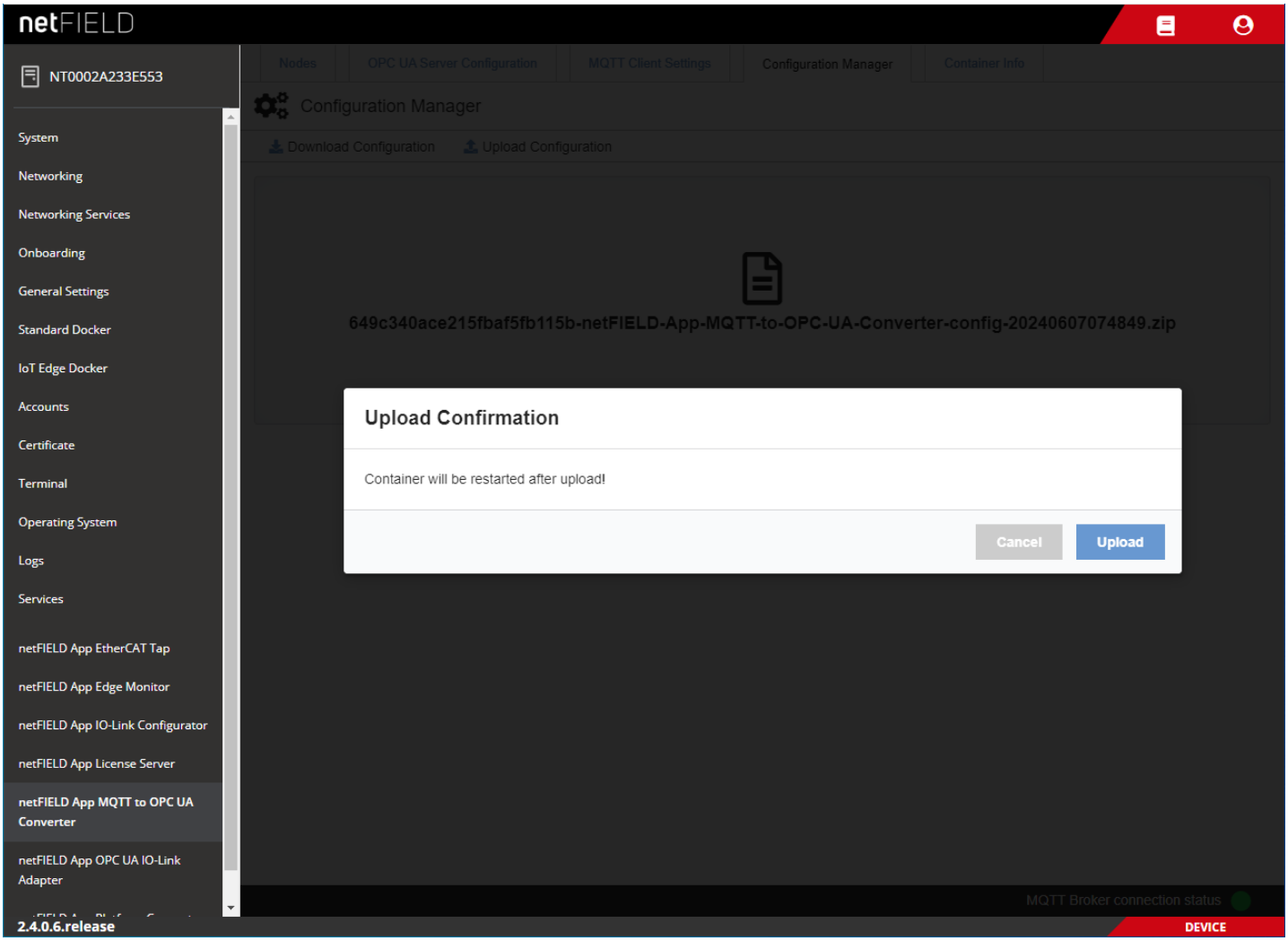


Figure 10. Upload Configuration

5.6 Container Info

The Container Info tab shows general information about the container.

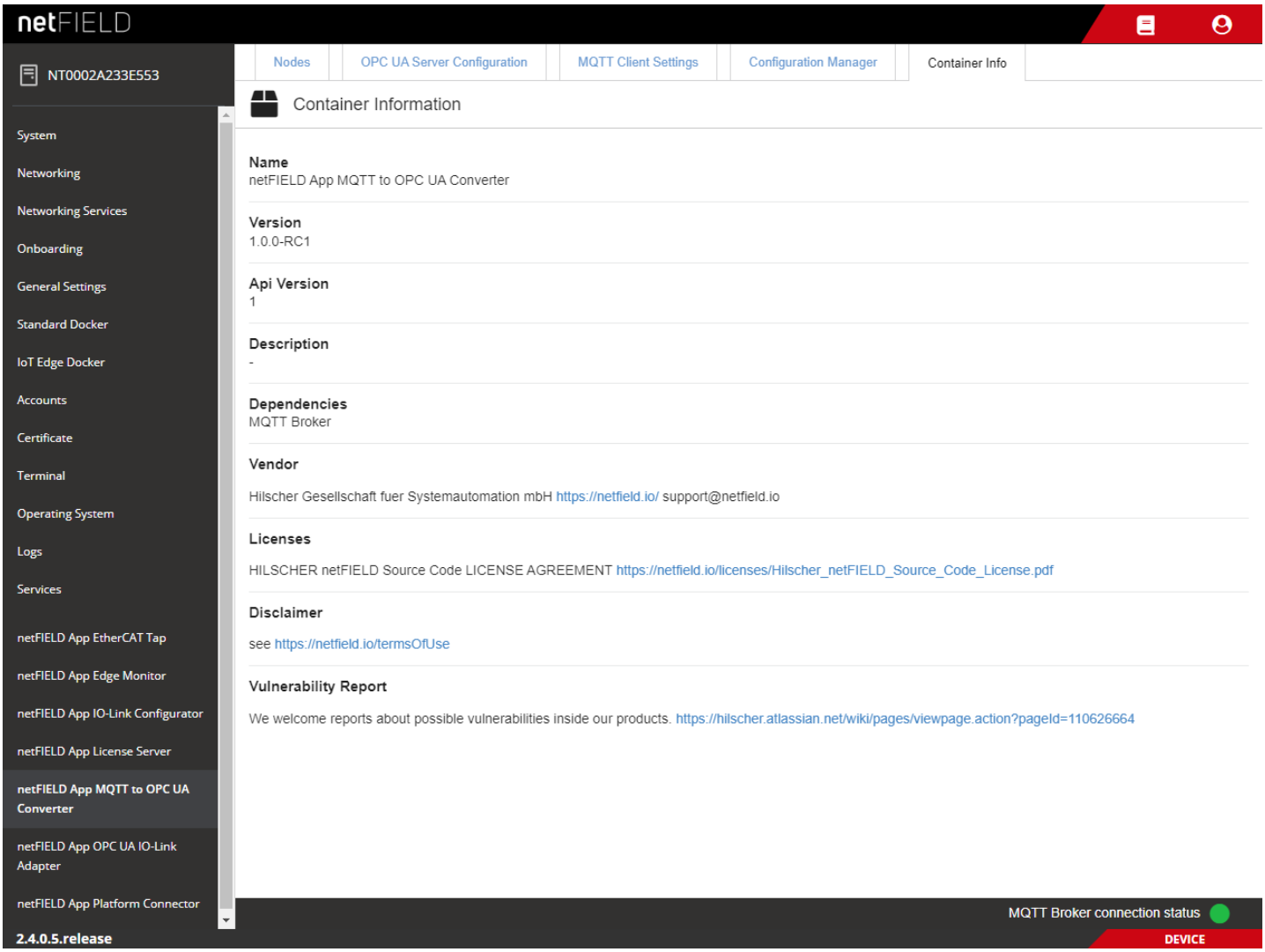


Figure 11. Container Info

Category	Description
Name	Container name
Version	Container software version
API Version	REST API version of the container
Description	Brief description of the function of the container
Dependencies	Other containers or components required for proper operation of the container
Vendor	Vendor of container
Licenses	Name of the software license(s), under which the container was published
Disclaimer	Path/link to the software license(s)
Vulnerability Report	Path/link to the Hilscher Vulnerability Handling & Management web page

Table 7. Container Info tab

6 Good to Know

6.1 Using SSL/TLS encryption (optional)

Please note the following if you intend to use SSL/TLS encryption:

The certificates and key files that the embedded MQTT client of the netFIELD App MQTT to OPC UA Converter container needs for establishing a secure SSL/TLS connection to the MQTT broker are not managed by the OPC UA Server container app itself. Instead, they are to be stored on the Edge Device and mapped into the container from the netFIELD OS. For this mapping, the following standard directories are mapped into the container when you use the default Create Options in the netFIELD Portal:

`/etc/ssl/`

`/usr/share/ca-certificates/`

NOTE If you require different directories for your use case, you may change the mapping of these "bind mounts" in the default Create Options of the container in the netFIELD Portal (see section Create Options).

As a user, you can store your required keys and certificates in these directories. By selecting the SSL / TLS option on the MQTT Client Settings page (see section MQTT Client Settings), you can allow the embedded MQTT client of the OPC UA Server app container to use these files for establishing its secure SSL/TLS connection.

Note that these keys and certificates must be stored in PEM format (a specific file format for storing this kind of data) and that you have to specify the full path to the appropriate PEM file in the corresponding fields of the MQTT Client Settings page. For example:

File name and path to private key in PEM format: `/etc/ssl/private/client-key.pem`

File name and path to certificate chains in PEM format: `/etc/ssl/services/client-cert.pem`

Override the trusted CA certificates in PEM format: `/etc/ssl/services/ca-cert.pem`

IMPORTANT If you intend to use more than one "secure" MQTT broker (as listed in the Server URIs field), and thus require several different certificates, you have to store them *in one single* PEM file. This is because it is not possible to specify a list of multiple paths to separate PEM files for individual brokers.

6.2 Container configuration data storage

The configuration data of the container is stored in the `netfield-app-mqtt-to-opc-ua-converter-data` Docker volume.

Your whole application configuration data - such as your MQTT configuration, activated publishers etc. - is stored here independently of the run state or the version of your netFIELD App MQTT to OPC UA Converter container. When you stop and restart the container, the configuration will be loaded from this volume again.

IMPORTANT **Automatic Upgrading**

The configuration data in this volume will be automatically migrated to the latest version when you deploy a newer version of the netFIELD App MQTT to OPC UA Converter.

CAUTION **No backwards compatibility**

Only upgrading towards *higher* versions is possible. If you try to start a lower container version with a newer configuration volume, the configuration will not be loaded, but will be cleared instead.

7 Appendix

Appendix A: Content listing

7.A.1 List of tables

7.A.2 List of figures

Appendix B: Legal Notes

Copyright

Important notes

Utmost care was/is given in the preparation of the documentation at hand consisting of a user's manual, operating manual and any other document type and accompanying texts. However, errors cannot be ruled out. Therefore, we cannot assume any guarantee or legal responsibility for erroneous information or liability of any kind. You are hereby made aware that descriptions found in the user's manual, the accompanying texts and the documentation neither represent a guarantee nor any indication on proper use as stipulated in the agreement or a promised attribute. It cannot be ruled out that the user's manual, the accompanying texts and the documentation do not completely match the described attributes, standards or any other data for the delivered product. A warranty or guarantee with respect to the correctness or accuracy of the information is not assumed.

We reserve the right to modify our products and the specifications for such as well as the corresponding documentation in the form of a user's manual, operating manual and/or any other document types and accompanying texts at any time and without notice without being required to notify of said modification. Changes shall be taken into account in future manuals and do not represent an obligation of any kind, in particular there shall be no right to have delivered documents revised. The manual delivered with the product shall apply.

Under no circumstances shall Hilscher Gesellschaft für Systemautomation mbH be liable for direct, indirect, ancillary or subsequent damage, or for any loss of income, which may arise after use of the information contained herein.

Liability disclaimer

The hardware and/or software was created and tested by Hilscher Gesellschaft für Systemautomation mbH with utmost care and is made available as is. No warranty can be assumed for the performance or flawlessness of the hardware and/or software under all application conditions and scenarios and the work results achieved by the user when using the hardware and/or software. Liability for any damage that may have occurred as a result of using the hardware and/or software or the corresponding documents shall be limited to an event involving willful intent or a grossly negligent violation of a fundamental contractual obligation. However, the right to assert damages due to a violation of a fundamental contractual obligation shall be limited to contract-typical foreseeable damage.

It is hereby expressly agreed upon in particular that any use or utilization of the hardware and/or software in connection with

- Flight control systems in aviation and aerospace;
- Nuclear fission processes in nuclear power plants;
- Medical devices used for life support and
- Vehicle control systems used in passenger transport

shall be excluded. Use of the hardware and/or software in any of the following areas is strictly prohibited:

- For military purposes or in weaponry;
- For designing, engineering, maintaining or operating nuclear systems;
- In flight safety systems, aviation and flight telecommunications systems;
- In life-support systems;
- In systems in which any malfunction in the hardware and/or software may result in physical injuries or fatalities.

You are hereby made aware that the hardware and/or software was not created for use in hazardous environments, which require fail-safe control mechanisms. Use of the hardware and/or software in this kind of environment shall be at your own risk; any liability for damage or loss due to impermissible use shall be excluded.

Warranty

Hilscher Gesellschaft für Systemautomation mbH hereby guarantees that the software shall run without errors in accordance with the requirements listed in the specifications and that there were no defects on the date of acceptance. The warranty period shall be 12 months commencing as of the date of acceptance or purchase (with express declaration or implied, by customer's conclusive behavior, e.g. putting into operation permanently).

The warranty obligation for equipment (hardware) we produce is 36 months, calculated as of the date of delivery ex works. The aforementioned provisions shall not apply if longer warranty periods are mandatory by law pursuant to Section 438 (1.2) BGB, Section 479 (1) BGB and Section 634a (1) BGB [Bürgerliches Gesetzbuch; German Civil Code] If, despite of all due care taken, the delivered product should have a defect, which already existed at the time of the transfer of risk, it shall be at our discretion to either repair the product or to deliver a replacement product, subject to timely notification of defect.

The warranty obligation shall not apply if the notification of defect is not asserted promptly, if the purchaser or third party has tampered with the products, if the defect is the result of natural wear, was caused by unfavorable operating conditions or is due to violations against our operating regulations or against rules of good electrical engineering practice, or if our request to return the defective object is not promptly complied with.

Costs of support, maintenance, customization and product care

Please be advised that any subsequent improvement shall only be free of charge if a defect is found. Any form of technical support, maintenance and customization is not a warranty service, but instead shall be charged extra.

Additional guarantees

Although the hardware and software was developed and tested in-depth with greatest care, Hilscher Gesellschaft für Systemautomation mbH shall not assume any guarantee for the suitability thereof for any purpose that was not confirmed in writing. No guarantee can be granted whereby the hardware and software satisfies your requirements, or the use of the hardware and/or software is uninterruptable or the hardware and/or software is fault-free.

It cannot be guaranteed that patents and/or ownership privileges have not been infringed upon or violated or that the products are free from third-party influence. No additional guarantees or promises shall be made as to whether the product is market current, free from deficiency in title, or can be integrated or is usable for specific purposes, unless such guarantees or promises are required under existing law and cannot be restricted.

Confidentiality

The customer hereby expressly acknowledges that this document contains trade secrets, information protected by copyright and other patent and ownership privileges as well as any related rights of Hilscher Gesellschaft für Systemautomation mbH. The customer agrees to treat as confidential all of the information made available to customer by Hilscher Gesellschaft für Systemautomation mbH and rights, which were disclosed by Hilscher Gesellschaft für Systemautomation mbH and that were made accessible as well as the terms and conditions of this agreement itself.

The parties hereby agree to one another that the information that each party receives from the other party respectively is and shall remain the intellectual property of said other party, unless provided for otherwise in a contractual agreement.

The customer must not allow any third party to become knowledgeable of this expertise and shall only provide knowledge thereof to authorized users as appropriate and necessary. Companies associated with the customer shall not be deemed third parties. The customer must obligate authorized users to confidentiality. The customer should only use the confidential information in connection with the performances specified in this agreement.

The customer must not use this confidential information to his own advantage or for his own purposes or rather to the advantage or for the purpose of a third party, nor must it be used for commercial purposes and this confidential information must only be used to the extent provided for in this agreement or otherwise to the extent as expressly authorized by the disclosing party in written form. The customer has the right, subject to the obligation to confidentiality, to disclose the terms and conditions of this agreement directly to his legal and financial consultants as would be required for the customer's normal business operation.

Export provisions

The delivered product (including technical data) is subject to the legal export and/or import laws as well as any associated regulations of various countries, especially such laws applicable in Germany and in the United States. The products / hardware / software must not be exported into such countries for which export is prohibited under US American export control laws and its supplementary provisions. You hereby agree to strictly follow the regulations and to yourself be responsible for observing them. You are hereby made aware that you may be required to obtain governmental approval to export, reexport or import the product.

Appendix C: Contacts

Germany
Hilscher Gesellschaft für Systemautomation mbH
Rheinstrasse 15
65795 Hattersheim
Phone: +49 (0) 6190 9907-0
Fax: +49 (0) 6190 9907-50
E-Mail: info@hilscher.com

Support
Phone: +49 (0) 6190 9907-990
E-Mail: de.support@hilscher.com

China
Hilscher Systemautomation (Shanghai) Co. Ltd.
200010 Shanghai
Phone: +86 (0) 21-6355-5161
E-Mail: info@hilscher.cn

Support
Phone: +86 (0) 21-6355-5161
E-Mail: cn.support@hilscher.com

France
Hilscher France S.a.r.l.
69800 Saint Priest
Phone: +33 (0) 4 72 37 98 40
E-Mail: info@hilscher.fr

Support
Phone: +33 (0) 4 72 37 98 40
E-Mail: fr.support@hilscher.com

India
Hilscher India Pvt. Ltd.
Pune, Delhi, Mumbai, Bangalore
Phone: +91 8888 750 777
E-Mail: info@hilscher.in

Support
Phone: +91 8108884011
E-Mail: info@hilscher.in

Austria
Hilscher Austria GmbH
4020 Linz
Phone: +43 732 931 675-0
E-Mail: sales.at@hilscher.com

Support
Phone: +43 732 931 675-0
E-Mail: at.support@hilscher.com

USA
Hilscher North America, Inc.
Lisle, IL 60532
Phone: +1 630-505-5301
E-Mail: info@hilscher.us

Support
Phone: +1 630-505-5301
E-Mail: us.support@hilscher.com

Japan
Hilscher Japan KK
Tokyo, 160-0022
Phone: +81 (0) 3-5362-0521
E-Mail: info@hilscher.jp

Support
Phone: +81 (0) 3-5362-0521
E-Mail: jp.support@hilscher.com

Republic of Korea
Hilscher Korea Inc.
13494, Seongnam, Gyeonggi
Phone: +82 (0) 31-739-8361
E-Mail: info@hilscher.kr

Support
Phone: +82 (0) 31-739-8363
E-Mail: kr.support@hilscher.com

Switzerland
Hilscher Swiss GmbH
4500 Solothurn
Phone: +41 (0) 32 623 6633
E-Mail: info@hilscher.ch

Support
Phone: +41 (0) 32 623 6633
E-Mail: ch.support@hilscher.com

Italy
Hilscher Italia S.r.l.
20090 Vimodrone (MI)
Phone: +39 02 25007068
E-Mail: info@hilscher.it

Support
Phone: +39 02 25007068
E-Mail: it.support@hilscher.com