Helmholz®
COMPATIBLE WITH YOU

PN/MQTT Coupler

Manual

Version 4 | 4.2.2022 | for firmware V1.08 and above

Link to newest version of manual

Helmholz GmbH & Co. KG | Hannberger Weg 2 | D-91091 Großenseebach | Germany
Phone +49 9135 7380-0 | Fax +49 9135 7380-110 | info@helmholz.de | www.helmholz.com

Notes

To download the latest version of this manual, please visit our website at www.helmholz.de.

We welcome all ideas and suggestions.

Helmholz GmbH & Co. KG

Hannberger Weg 2 | 91091 Großenseebach

All trademarks shown or mentioned in this document are the property of their respective owners or manufacturers. The representation and naming serve exclusively to explain the use and setting options of the products documented here.

Revision Record:

Version	Date	Change
1	12.4.2021	First version
2	18.6.2021	Update for firmware V1.04 Microsoft Azure example added
3	26.10.2021	Correction of dimensions Updates for firmware V1.06 Update of security recommendations
4	4.2.2022	Updates for firmware V1.08: Payload Editor for subscribing modules AWS and HiveMQ application examples updated

Contents

1 General

This operating manual applies only to devices, assemblies, software, and services of Helmholz GmbH & Co. KG.

1.1 Structure of the manual

This manual is divided into 19 sections.

Section 1 contains general information and safety instructions.

Section 2 refers to Security Recommendations.

Section 3 explains the system overview and features of the product.

Section 4+5 explain the mounting and electrical connection of the product.

Sections 6-11 explain the configuration and programming of the product.

Sections 12+13 describe functions for maintaining and diagnosing the product.

Sections 14-18 contain application examples.

The technical data can be found in section 19.

1.2 Target audience for this manual

This description is only intended for trained personnel qualified in control and automation engineering who are familiar with the applicable national standards. For installation, commissioning, and operation of the components, compliance with the instructions and explanations in this operating manual is essential.

WARNING

Configuration, execution, and operating errors can interfere with the proper operation of the device and result in personal injury, as well as material or environmental damage. Only suitably qualified personnel may operate the devices!

Qualified personnel must ensure that the application and use of the products described meet all the safety requirements, including all relevant laws, regulations, provisions, and standards.

1.3 Safety instructions

The safety instructions must be observed in order to prevent harm to living creatures, material goods, and the environment. The safety notes indicate possible hazards and provide information about how hazardous situations can be prevented.

1.4 Note symbols and signal words

HAZARD

If the hazard warning is ignored, there is an imminent danger to life and health of people from electrical voltage.

WARNING

If the warning is ignored, there is a probable danger to life and health of people.

CAUTION

If the caution note is ignored, people can be injured or harmed.

ATTENTION

Draws attention to sources of error that can damage equipment or the environment.

NOTE

Gives an indication for better understanding or preventing errors.

1.5 Intended use

The "PN/MQTT Coupler" enables data exchange between a PROFINET network and MQTT brokers.

All components are supplied with a factory hardware and software configuration. The user must carry out the hardware and software configuration for the conditions of use. Modifications to hardware or software configurations which are beyond the documented options are not permitted and nullify the liability of Helmholz GmbH & Co. KG.

⚠ WARNING

The device may not be used as the only means for preventing hazardous situations on machinery and systems.

Problem-free and safe operation of the device presumes proper transport, storage, setup, assembly, installation, commissioning, operation, and maintenance.

The ambient conditions provided in the technical specifications must be adhered to.

The device has a protection rating of IP20 and must be installed in an electrical operating room or a control box/cabinet in order to protect it against environmental influences. To prevent unauthorized access, the doors of control boxes/cabinets must be closed and possibly locked during operation.

1.6 Improper use

⚠ WARNING

The consequences of improper use may include personal injuries of the user or third parties as well as property damage to the control system, the product, or the environment. Use the PN/MQTT Coupler only as intended!

1.7 Liability

The contents of this manual are subject to technical changes resulting from the continuous development of products of Helmholz GmbH & Co. KG. In the event that this manual contains technical or clerical errors, we reserve the right to make changes at any time without notice.

No claims for modification of delivered products can be asserted based on the information, illustrations, and descriptions in this documentation. Beyond the instructions contained in the operating manual, the applicable national and international standards and regulations must also be observed in any case.

1.7.1 Disclaimer of liability

Helmholz GmbH &Co. KG is not liable for damages if these were caused by use or application of products that was improper or not as intended.

Helmholz GmbH & Co. KG assumes no responsibility for any printing errors or other inaccuracies that may appear in the operating manual unless there are serious errors about which Helmholz GmbH & Co. KG was already demonstrably aware.

Beyond the instructions contained in the operating manual, the applicable national and international standards and regulations must also be observed in any case.

Helmholz GmbH & CO. KG is not liable for damage caused by software that is running on the user's equipment which compromises, damages, or infects additional equipment or processes through the remote maintenance connection and which triggers or permits unwanted data transfer.

1.7.2 Warranty

Report any defects to the manufacturer immediately after discovery of the defect.

The warranty is not valid in case of:

- Failure to observe these operating instructions
- Use of the device that is not as intended
- Improper work on and with the device
- Operating errors
- Unauthorized modifications to the device

The agreements met upon contract conclusion under "General Terms and Conditions of Helmholz GmbH & Co. KG" apply.

1.8 Open Source

Among other things, our products contain open source software. This software is subject to the relevant license terms. The relevant license terms, including a copy of the full license text, are downloadable from the product website. They are also provided in our download area of the respective products at www.helmholz.de.

Furthermore, we offer to send the complete corresponding source code of the respective open source software to you and to any third party as a DVD upon your request for a contribution towards expenses of Euro 10.00. This offer is valid for a period of three years. This offer is valid for a period of three years, calculated from the delivery of the product.

2 Security recommendations

Managed switches are network infrastructure components, and thus an important element in the security considerations of a system or network. When using the PN/MQTT Coupler, therefore please consider the following recommendations in order to prohibit unauthorized access to plants and systems.

General:

- Ensure at regular intervals that all relevant components fulfill these recommendations and possibly any other internal security guidelines.
- Evaluate your system holistically with a view to security. Use a cell protection concepts (“defense-in-depth”) with corresponding products, such as the WALL IE.
- Regularly inform yourself about security threats for all your components

Physical access:

- Limit physical access to components of relevance to security to qualified personnel.

Security of the software:

- Always keep the firmware of all communications components up to date.
- Inform yourself regularly of firmware updates for the product.
- Only activate protocols and functions you really need
- If possible, always use those variants of protocols that provide more security

Passwords:

- Define rules and roles for usage of the devices and the awarding of passwords
- Change standard passwords
- Only use strong passwords. Avoid weak passwords like, for example, “password1”, “123456789”, or similar.
- Ensure that all passwords are inaccessible to unauthorized personnel.
- Don’t use one password for various users and systems.

Helmholz is a member of the CERT@VDE. In addition to our technical newsletter, we communicate our security-relevant updates, patches and advisories to you as a user of Helmholz products. Find out more and use the services and database of the **CERT@VDE** to make your systems secure and keep them secure.

The Helmholz "**Product Security Incident Response Team**" **(PSIRT)** supports you proactively to protect your machines as best as possible in the context of industrial communication. Whenever new potential threats occur or are reported to us, we evaluate and process them immediately and provide you with recommended actions, patches and updates as quickly as possible to reduce the risk to a minimum.

You can help too: Report any product incidents to our **Product Security Incident Response Team** at psirt@helmholz.de or support@helmholz.de.

You can find more information on the topic of security here, for example:

- CERT@VDE
- Sichere-industrie.de
- Bundesamt für Sicherheit in der Informationstechnik (BSI)
- Allianz für Cyber-Sicherheit

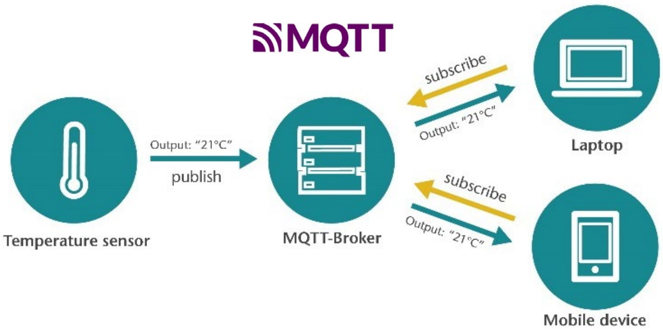
3 System overview

3.1 General/area of application

The MQTT protocol has become established in recent years as a simple transmission protocol for messages in the IoT world. MQTT stands for "Message Queue Telemetry Transport Protocol" and is an OASIS standard. Information on the MQTT protocol can be found here: mqtt.org

The MQTT protocol always uses a central broker for communication between devices, which receives messages from e.g. a sensor and forwards them to interested devices, e.g. a control unit.

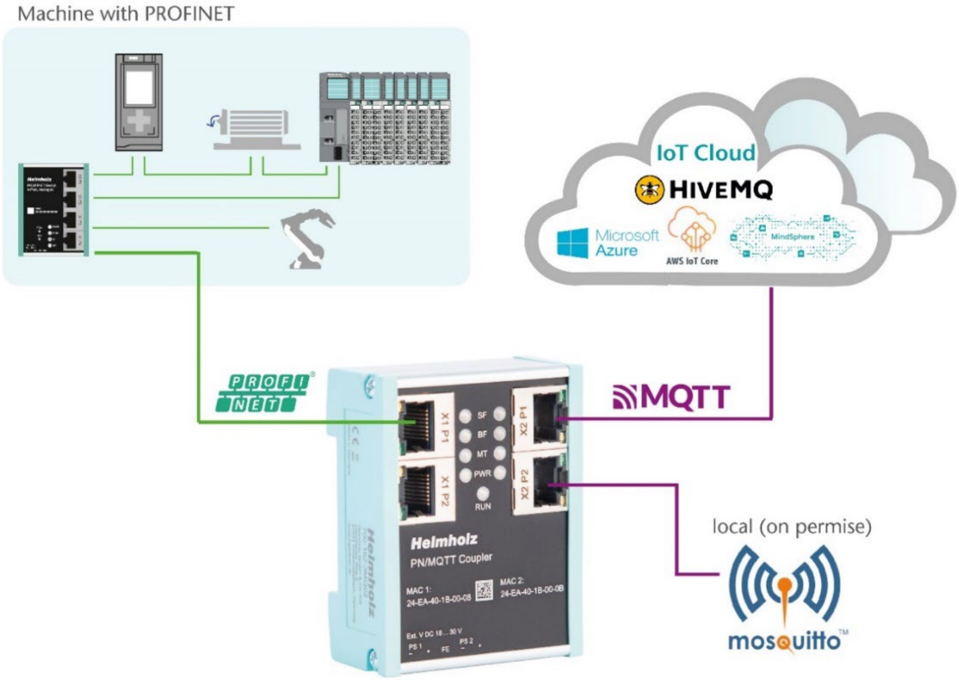
When a sensor sends data to the broker, this is called "publish". If a device needs data, it must “subscribe” to the broker. The broker delivers the data to all subscribers when new data has arrived from the publisher.



Data is always transmitted under a freely definable identifier - the "Topic". The "Topic" is a descriptive text, e.g. "Temperature". In order to distinguish between different topics of the same type, groupings are used, e.g. "Living room/temperature". The groupings are divided by a slash ('/'). Thus the data can be mapped into more complex structures: "Upper floor/Living room/Temperature" or "Upper floor/Living room/Humidity".

The data delivered via MQTT can be transmitted in binary format, text format or structured in JSON format.

With the PN/MQTT Coupler a transfer of values between a PROFINET controller (PLC) and a MQTT broker is possible. It is possible to send values from the PLC via PROFINET to a broker ("Publish") as well as to subscribe values from a MQTT broker and receive them in the PLC via PROFINET ("Subscribe").



The integration into the PLC engineering tool is enabled by a GSDML file, an additional extra configuration software is not necessary. The configuration of the I/O data to be exchanged is done in the Siemens engineering tool. All settings for the MQTT connection can be done on the web page of the device.

MQTT brokers can be connected both locally ("On premise") and via the Internet ("Cloud"). A local broker can be operated, for example, with the open-source software "Mosquitto" in the company network on a PC/server or also on a small computer, such as a Raspberry PI.

NOTE The PN/MQTT coupler can only establish a connection with one broker! If the data is also to be distributed to other brokers, the connection must be established between the brokers (Multi Broker).

In the cloud, IoT systems such as HiveMQ, Amazon IoT, Microsoft Azure or Siemens Mind-sphere *(in preparation)* can be connected directly. For a description of how to connect to the various cloud systems, see later in this manual or ask support.

3.2 PN/MQTT Coupler Features

The "PN/MQTT Coupler" has the following features:

- PROFINET IO Device as of IEC 61158-6-10
- Up to 1024 bytes of input and output data
- Supports OASIS MQTT standards V3.1.1 and V5
- Easy assignment of IO data via GSDML file
- Up to 100 values can be configured for transmission at the same time (100 slots)
- Flexible configuration via web browser
- Separate networks for PROFINET and MQTT connection
- Connection to brokers in the local network or directly with the "cloud”
- Authentication (password, certificate) and encryption (TLS)
- Supports AWS IoT, Microsoft Azure, HiveMQ, IBM Watson, Google IoT, Siemens Mindsphere (in preparation)
- Very compact design for DIN rail mounting
- Redundant power supply
- Galvanic isolation of the networks

4 Installation and removal

4.1 Access restriction

The modules are open operating equipment and must only be installed in electrical equipment rooms, cabinets, or housings.

Access to the electrical equipment rooms, cabinets, or housings must only be possible using a tool or key, and access should only be granted to trained or authorized personnel.

4.2 Mounting and minimum distances

The PN/MQTT Coupler can be mounted on a DIN rail and installed in any position. It is recommended to keep minimum distances when mounting. By keeping the minimum distances

- the modules can be mounted or dismantled without having to dismantle other parts of the system.
- there is enough space to connect all existing connections and contacting possibilities with commercially available accessories.
- There is space for any necessary cable routing.

ATTENTION

Installation must be carried out in accordance with VDE 0100/IEC 364 and applicable national standards. The device has protection level IP20. If a higher degree of protection is required, it must be installed in an enclosure or a control cabinet.

4.3 Electrical installation

Observe the regional safety regulations.

4.4 Protection against electrostatic discharges

To prevent damage through electrostatic discharges, the following safety measures are to be followed during assembly and service work:

- Never place components and modules directly on plastic items (such as polystyrene, PE film) or in their vicinity.
- Before starting work, touch the grounded housing to discharge static electricity.
- Only work with discharged tools.
- Do not touch components and assemblies on contacts.

4.5 EMC protection

To ensure electromagnetic compatibility (EMC) in your control cabinets in electrically harsh environments, the known rules of EMC-compliant configuration are to be observed in the design and construction.

ATTENTION

Observe all standards, regulations and rules regarding shielding when setting up the system and laying the necessary cables. Strictly adhere to the corresponding writings of the PROFIBUS user organization for setting up PROFINET.
Errors in the shielding can lead to malfunctions or even failure of the system.

4.6 Operation

Operate the device only in flawless condition. The permissible operating conditions and performance limits must be adhered to.

Retrofits, changes, or modifications to the device are strictly forbidden.

The device is a piece of operating equipment intended for use in industrial plants. During operation, all covers on the unit and the installation must be closed in order to ensure protection against contact

ATTENTION

When the PN/MQTT Coupler is switched off, bus connections are interrupted! Before starting any work on the device, make sure that no impermissible interference occurs in connected systems when the bus connections are interrupted.

4.7 Recycling / WEEE

The company Helmholz GmbH & Co. KG is registered as a manufacturer with the HELMHOLZ brand and the device type "Small devices of information and telecommunications technology for exclusive use in households other than private households" as well as the following registration data:

Helmholz GmbH & Co. KG,
Location / Headquarters: 91091 Großenseebach,
Address: Hannberger Weg 2,
Name of authorized representative: Carsten Bokholt,

Registration number: **DE 44315750**

The electrical devices described in this document are to be recycled. According to Directive 2012/19/EU on waste electrical and electronic equipment (WEEE), they must not be disposed of by municipal waste disposal companies.

5 Connection

5.1 Power supply

The PN/MQTT coupler must be supplied with DC 24 V at the wide-range input DC 18 ... 30 V via the supplied connector plug. The power supply is redundant, at least one supply path PS 1 or PS 2 must be connected.

NOTE

The housing of the PN/MQTT Coupler is not grounded. Please connect the functional earth terminal (FE) of the PN/MQTT Coupler properly to the reference gantry.

5.2 Network

The left RJ45 sockets "X1 P1" and "X1 P2" are used to connect the PROFINET network, the right RJ45 sockets "X2 P1" and "X2 P2" are used to connect the Ethernet network in which the MQTT broker is accessible. The ports X1 P1 and X1 P2, as well as X2 P1 and X2 P2 are each internally connected to a switch.

The interfaces X1 and X2 are logically separate networks and not physically connected. Thus a clear separation between the machine data (PROFINET) and the MQTT data connection is possible. A network penetration with other functions by the PN/MQTT coupler is not possible.

The configured values are exchanged in the PN/MQTT coupler only as IO data between both network sides.

X1: PROFINET-Stack	Internal Memory	X2: MQTT Client
Outputs	→	Publish
Inputs	←	Subscribe

NOTE

If the MQTT broker needs to be placed in the same network (subnet) as the PROFINET PLC, the network X2 can be configured in the same subnet as the network X1. The interface X2 then needs its own IP address and must be connected to the network X1 with an Ethernet cable.

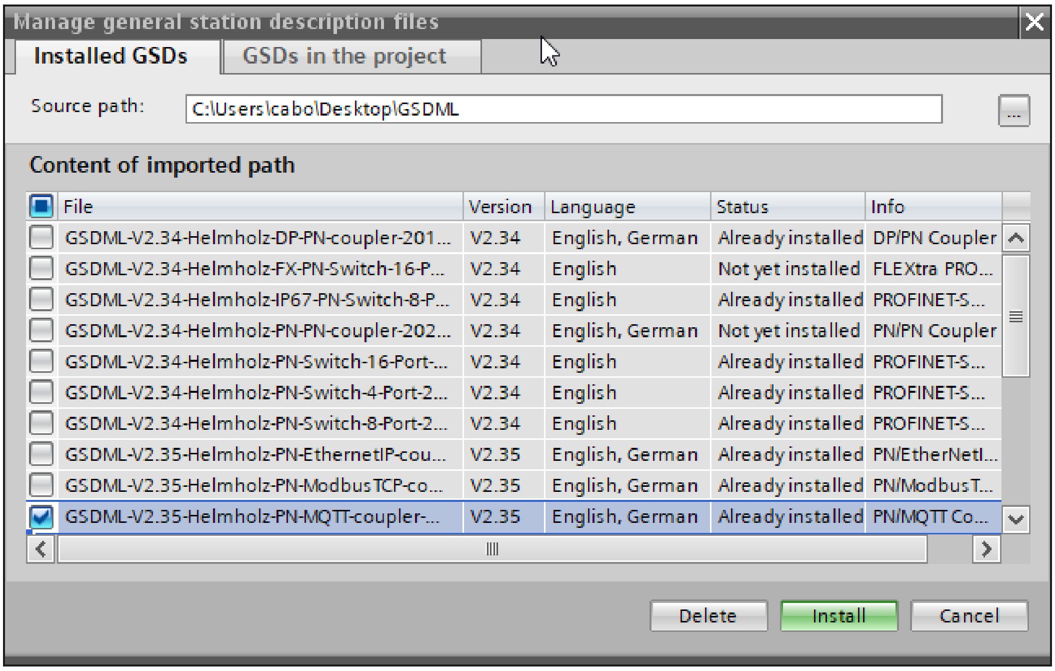
5.3 network connector

Pin	signal	RJ45 connector	color	Wire pair
1	TD+	Transmission Data +	Yellow	1
2	TD-	Transmission Data -	Orange	1
3	RD+	Receive Data +	White	2
4	-	-	-	-
5	-	-	-	-
6	RD-	Receive Data -	Blue	2
7	-	-	-	-

8	-	-	-	-

6 Install GSDML file

Please download the GSDML file ("GSDML-V2.35-Helmholz-PN-MQTT-coupler-____.xml") at www.helmholz.de or scan the QR code. Install the GSDML file in the "Tools" / "Manage device description file (GSD)" menu in the TIA Portal.



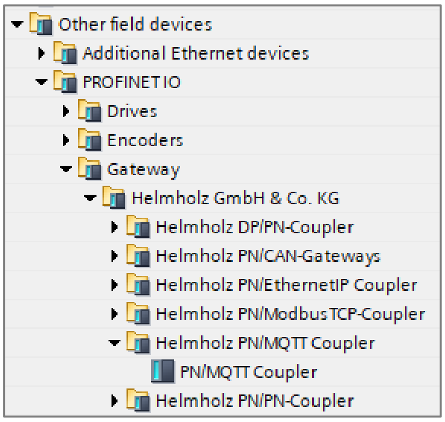
Manage general station description files

Source path: C:\Users\cabo\Desktop\GSDML

Content of imported path

☑	File	Version	Language	Status	Info
□	GSDML-V2.34-Helmholz-DP-PN-coupler-201...	V2.34	English, German	Already installed	DP/PN Coupler
□	GSDML-V2.34-Helmholz-FX-PN-Switch-16-P...	V2.34	English	Not yet installed	FLEXtra PRO...
□	GSDML-V2.34-Helmholz-IP67-PN-Switch-8-P...	V2.34	English	Already installed	PROFINET-S...
□	GSDML-V2.34-Helmholz-PN-PN-coupler-202...	V2.34	English, German	Not yet installed	PN/PN Coupler
□	GSDML-V2.34-Helmholz-PN-Switch-16-Port-...	V2.34	English	Already installed	PROFINET-S...
□	GSDML-V2.34-Helmholz-PN-Switch-4-Port-2...	V2.34	English	Already installed	PROFINET-S...
□	GSDML-V2.34-Helmholz-PN-Switch-8-Port-2...	V2.34	English	Already installed	PROFINET-S...
□	GSDML-V2.35-Helmholz-PN-EthernetIP-cou...	V2.35	English, German	Already installed	PN/EtherNetI...
□	GSDML-V2.35-Helmholz-PN-ModbusTCP-co...	V2.35	English, German	Already installed	PN/ModbusT...
☑	GSDML-V2.35-Helmholz-PN-MQTT-coupler-...	V2.35	English, German	Already installed	PN/MQTT Co...

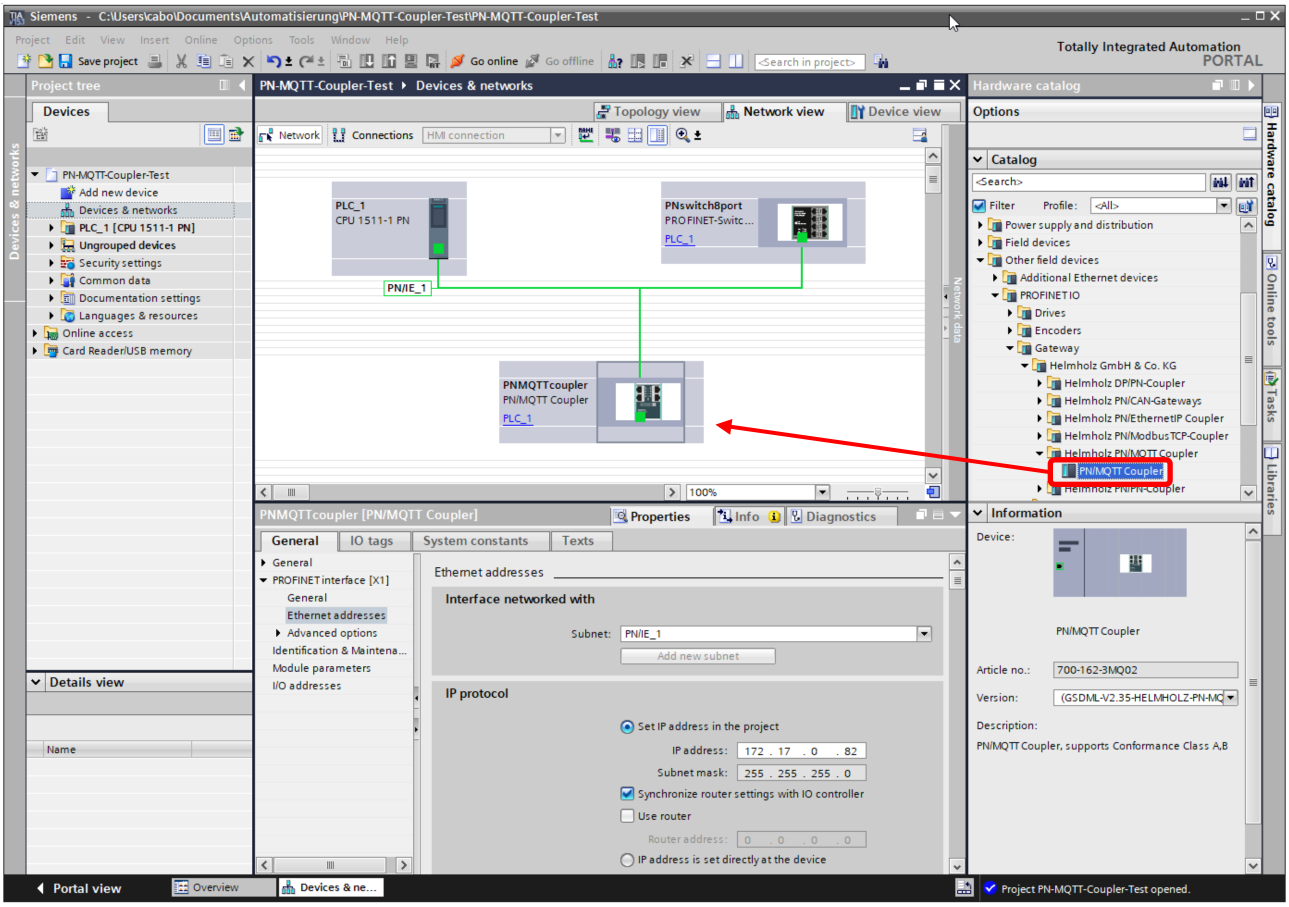
The PN/MQTT Coupler can be found in the hardware catalog at "Other field devices / PROFINET IO / Gateway / Helmholz GmbH & Co. KG".



- Other field devices
  - Additional Ethernet devices
  - PROFINET IO
    - Drives
    - Encoders
    - Gateway
      - Helmholz GmbH & Co. KG
        - Helmholz DP/PN-Coupler
        - Helmholz PN/CAN-Gateways
        - Helmholz PN/EthernetIP Coupler
        - Helmholz PN/ModbusTCP-Coupler
        - Helmholz PN/MQTT Coupler
          - PN/MQTT Coupler
        - Helmholz PN/PN-Coupler

7 Configuration in TIA Portal

Add the PN/MQTT coupler to the project and connect the coupler to the PROFINET network.



Ethernet addresses

Interface networked with

Subnet: PN/IE_1

Add new subnet

IP protocol

● Set IP address in the project

IP address: 172 . 17 . 0 . 82

Subnet mask: 255 . 255 . 255 . 0

☑ Synchronize router settings with IO controller

☐ Use router

Router address: 0 . 0 . 0 . 0

○ IP address is set directly at the device

Device:

PN/MQTT Coupler

Article no.: 700-162-3MQ02

Version: (GSDML-V2.35-HELMHOLZ-PN-MQ

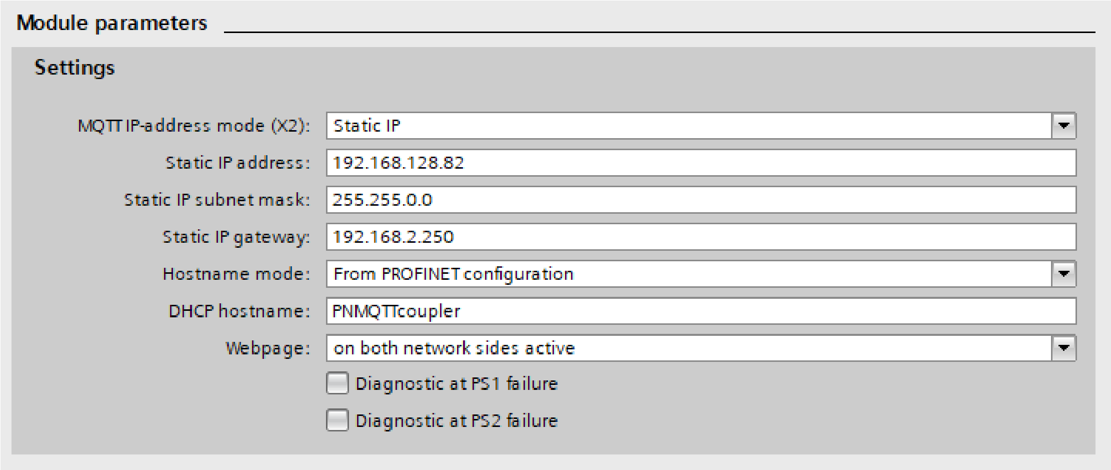
Description: PN/MQTT Coupler, supports Conformance Class A,B

Project PN-MQTT-Coupler-Test opened.

Name the device name and check the Ethernet address for the device.

7.1 Parameterization of the PN/MQTT coupler

The parameterization of the PN/MQTT coupler is performed via the PROFINET hardware configurator (e.g. TIA Portal). The PROFINET parameters and the I/O data to be transmitted via MQTT are defined here. The configuration of the MQTT network connection (connection to the MQTT broker) is set via the web page of the device.



Module parameters

Settings

MQTT IP-address mode (X2):	Static IP
Static IP address:	192.168.128.82
Static IP subnet mask:	255.255.0.0
Static IP gateway:	192.168.2.250
Hostname mode:	From PROFINET configuration
DHCP hostname:	PNMQTTcoupler
Webpage:	on both network sides active
	☐ Diagnostic at PS1 failure
	☐ Diagnostic at PS2 failure

MQTT IP-address mode (X2): Setting the IP address for the X2 network. Possible options:

"DHCP" = The PN/MQTT Coupler tries to get an IP address as well as a gateway and a DNS server in the network via a DHCP server.

"Static IP" = The address, subnet mask and date gateway can be fixed directly in the following fields. The setting of a DNS server can - if required - additionally be done on the web page.

"IP address from web page" = The IP settings of the X2 network can be carried out via the web page. At the first start-up, the PN/MQTT Coupler is only accessible in the X1 network (PROFINET). Only when the IP settings for the X2 network have been set there, the coupler is also accessible via X2 or can establish a connection to the MQTT broker.

Static IP address: If the address mode has been set to "Static IP", the static IP address of the X2 network can be specified here. This setting has no function with "DHCP" and "IP address from web page".

Static IP subnet mask: If the address mode has been set to "Static IP", the subnet mask of the X2 network can be specified here. This setting has no function with "DHCP" and "IP address from web page".

Static IP gateway: If the address mode has been set to "Static IP", the gateway of the X2 network can be specified here. This setting has no function with "DHCP" and "IP address from web page".

Hostname mode: "From PROFINET configuration" or "From webpage".

DHCP hostname: Hostname of the device, is used if the "Hostname mode" option "Take over from PROFINET configuration" was selected.

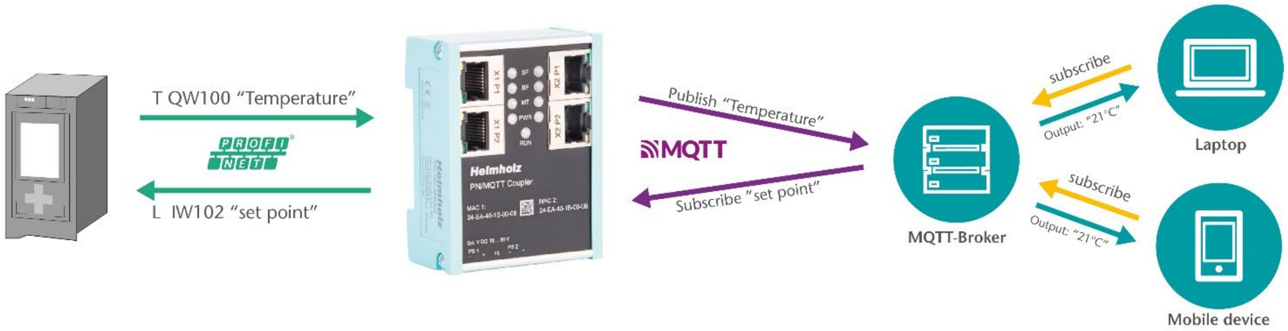
Webpage: On which network interfaces should the web page be displayed.

NOTE Please note in the commissioning phase in the PROFINET configuration to activate at least the web page on the PROFINET network side (X1) or "both network sides". Otherwise a complete configuration is not possible.

For security reasons it is advisable to switch off the web pages in the PROFINET configuration after commissioning or at least to switch off the web interface on the network side, which is connected in the WAN or Internet.

7.2 Operating principle of the PN/MQTT coupler

The data exchange between the PLC and the MQTT broker is organized via individual values. A value can be 1, 2 or 4 bytes in size and is in the I/O area of the PROFINET controller. Depending on the data direction, the value is writable to outputs (MQTT Publish) or readable from inputs (MQTT Subscribe).



Up to 100 different values can be exchanged between the PROFINET controller and the MQTT broker (100 slots). The values can be plugged as modules into the slots of the coupler as required.

A module always represents exactly one value, which is exchanged with the broker via its MQTT topic name. A value is usually sent via MQTT in a readable form (direct text or JSON formatted). For this reason, there are different representations for each value size (1, 2, 4, 8 bytes): hexadecimal, unsigned decimal, signed decimal or floating point.

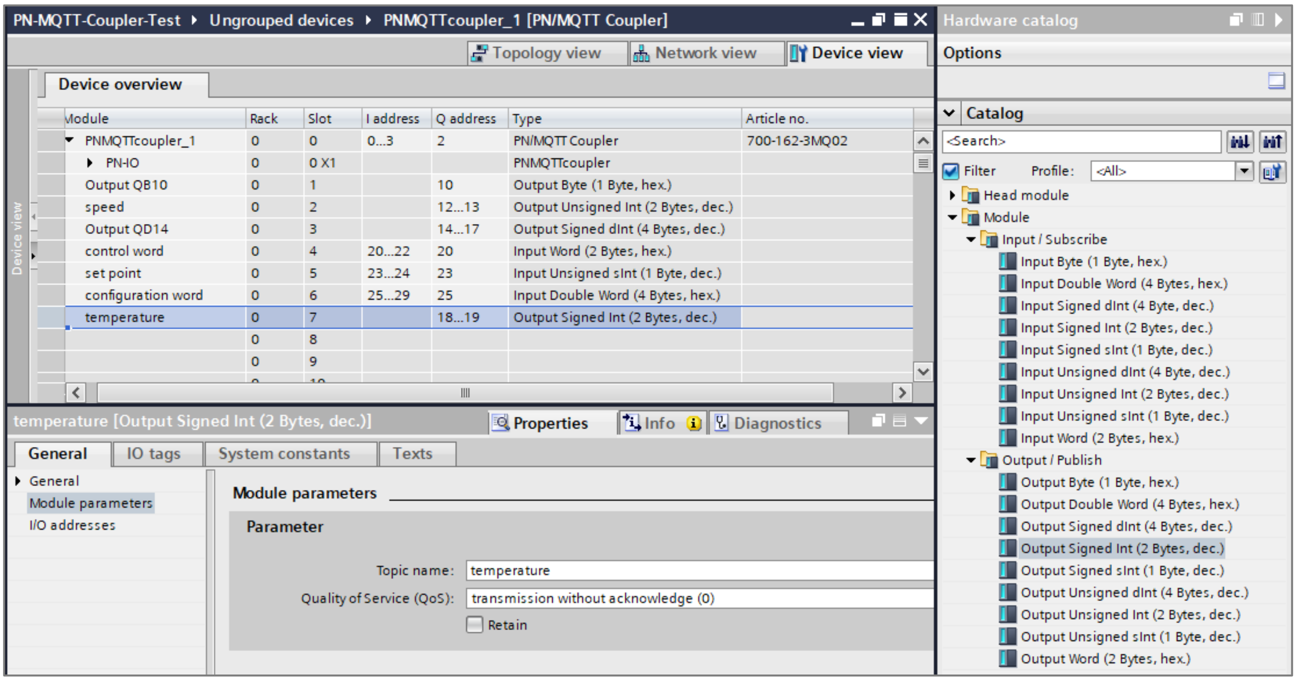
Configuration example:

Slot	Modul	EA	Type	Direction	Topic name (Exampe)	Value (Example)
1	Output Byte (1 Byte, hex.)	1 Byte Output	Byte	Publish →	“Output QB10”	“0x12”
2	Output Unsigned Int (2 Bytes, dez.)	2 Bytes Outputs	Unsigned Integer	Publish →	“Speed”	“65534”
3	Output Signed dInt (4 Bytes, dez.)	4 Bytes Outputs	Signed double Integer	Publish →	“Output QD14”	“-12345678”
4	Input Word (2 Bytes, hex.)	2 Bytes Inputs	Word	← Subscribe	„Control word “	„0xFFEE“
5	Input Unsigned sInt (1 Bytes, dez.)	1 Byte Inputs	Unsigned short Int	← Subscribe	„Set point “	„255“
6	Input Double Word (4 Byte, hex.)	4 Bytes Inputs	Double Word	← Subscribe	„Configuration “	„0x11223344“
7	Output Signed Int (2 Byte, dez.)	2 Bytes Outputs	Signed Integer	Publish →	„Temperature“	„25“
...						

Output modules are sent to the broker once after restarting the coupler and then after each change of the PLC value. If PLC values are sent that change very quickly, it is possible to specify a "publish interval" on the configuration web page.

The value of the input modules is initialized with 0 after (re-)starting the coupler and is permanently taken over into the input area when a new value is received via MQTT. A bit additionally indicates the reception of a value in the PLC.

The value of the input modules is initialized with 0 after restarting the coupler and is permanently taken over into the input area when a new value is received via MQTT. A bit additionally indicates the reception of a value in the PLC.


PN-MQTT-Coupler-Test ▸ Ungrouped devices ▸ PNMQTTcoupler_1 [PN/MQTT Coupler]
Topology view
Network view
Device view
Device view
Hardware catalog
Options
Catalog
<Search>
Filter
Profile:
<All>
Head module
Module
Input / Subscribe
Input Byte (1 Byte, hex.)
Input Double Word (4 Bytes, hex.)
Input Signed dInt (4 Byte, dec.)
Input Signed Int (2 Bytes, dec.)
Input Signed sInt (1 Byte, dec.)
Input Unsigned dInt (4 Byte, dec.)
Input Unsigned Int (2 Bytes, dec.)
Input Unsigned sInt (1 Byte, dec.)
Input Word (2 Bytes, hex.)
Output / Publish
Output Byte (1 Byte, hex.)
Output Double Word (4 Bytes, hex.)
Output Signed dInt (4 Bytes, dec.)
Output Signed Int (2 Bytes, dec.)
Output Signed sInt (1 Byte, dec.)
Output Unsigned dInt (4 Bytes, dec.)
Output Unsigned Int (2 Bytes, dec.)
Output Unsigned sInt (1 Byte, dec.)
Output Word (2 Bytes, hex.)
Properties
Info
Diagnostics
General
IO tags
System constants
Texts
General
Module parameters
I/O addresses

Device overview

Module	Rack	Slot	I address	Q address	Type	Article no.
PNMQTTcoupler_1	0	0	0...3	2	PN/MQTT Coupler	700-162-3MQ02
PN-IO	0	0 X1			PNMQTTcoupler	
Output QB10	0	1		10	Output Byte (1 Byte, hex.)	
speed	0	2		12...13	Output Unsigned Int (2 Bytes, dec.)	
Output QD14	0	3		14...17	Output Signed dInt (4 Bytes, dec.)	
control word	0	4	20...22	20	Input Word (2 Bytes, hex.)	
set point	0	5	23...24	23	Input Unsigned sInt (1 Byte, dec.)	
configuration word	0	6	25...29	25	Input Double Word (4 Bytes, hex.)	
temperature	0	7		18...19	Output Signed Int (2 Bytes, dec.)	
	0	8				
	0	9				

temperature [Output Signed Int (2 Bytes, dec.)]

Module parameters

Parameter

Topic name: temperature

Quality of Service (QoS): transmission without acknowledge (0)

☐ Retain

For each module, the **Topic name** must be defined unambiguously in the module parameters. The name can be chosen suitable to the symbolic name of the PLC value. Up to 40 characters are available.

As another parameter, the method of **Quality of Service (QoS)** of the topic can be specified.

Transmission without acknowledge (0): The topic is sent without an acknowledgement from the broker („fire-and-forget“)

Transmission with acknowledge (1): The topic is sent, and an acknowledgement ("PUBACK") is expected from the broker. If no acknowledgement is received, the topic is sent again.

Transmission with acknowledge and confirmation (2): Provides the guarantee that a message has been "delivered exactly once". In order to be able to keep this guarantee, MQTT uses a two-stage acknowledgement of receipt.

Retain: This option tells the broker to save the last message or value in the broker even if the connection to the MQTT client fails.

NOTE

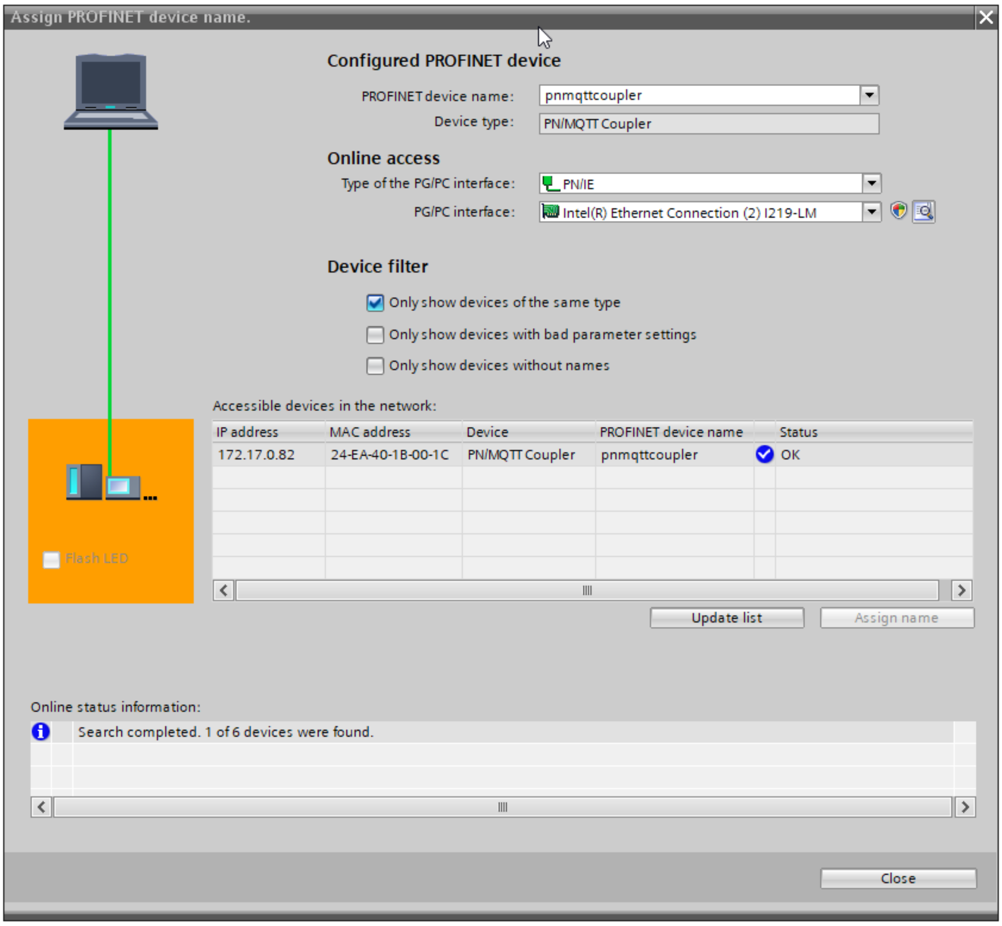
The format of the payload (value representation) of the MQTT messages is explained in chapter 8.3.

7.3 Assign a name to the PN/MQTT coupler

When the configuration of the PN/MQTT coupler is completed in the hardware configurator of the engineering tool, it can be imported into the PLC.

To enable the PN/MQTT Coupler to be found by the PROFINET controller, the PROFINET device name must be assigned to the PN/MQTT Coupler. To do this, use the "Assign device name" function, which you can access with the right mouse button or in the Online menu if the PN/MQTT Coupler is selected.

Use the "Update list" button to search the network for PROFINET stations. With "Assign name" the PROFINET device name can be assigned to the device.



Assign PROFINET device name.

Configured PROFINET device

PROFINET device name: pnmqttcoupler

Device type: PN/MQTT Coupler

Online access

Type of the PG/PC interface: PN/IE

PG/PC interface: Intel(R) Ethernet Connection (2) I219-LM

Device filter

☑ Only show devices of the same type

□ Only show devices with bad parameter settings

□ Only show devices without names

□ Flash LED

Accessible devices in the network:

IP address	MAC address	Device	PROFINET device name	Status
172.17.0.82	24-EA-40-1B-00-1C	PN/MQTT Coupler	pnmqttcoupler	✓ OK

Online status information:

Search completed. 1 of 6 devices were found.

The unique identification of the PN/MQTT coupler is guaranteed here by the MAC address of the device. The PROFINET MAC address can be read on the front of the PN/MQTT Coupler on the left-hand side at X1 ("MAC 1").

If the PN/MQTT coupler has received the correct PROFINET name, it is recognized and configured by the PLC. If the configuration is correct, the PROFINET "BF" LED should be off.

To set the PROFINET name, the Helmholz IPSet Tool can also be used, which can be downloaded free of charge from the Helmholz website or scan the following QR code to download the IPSet Tool.

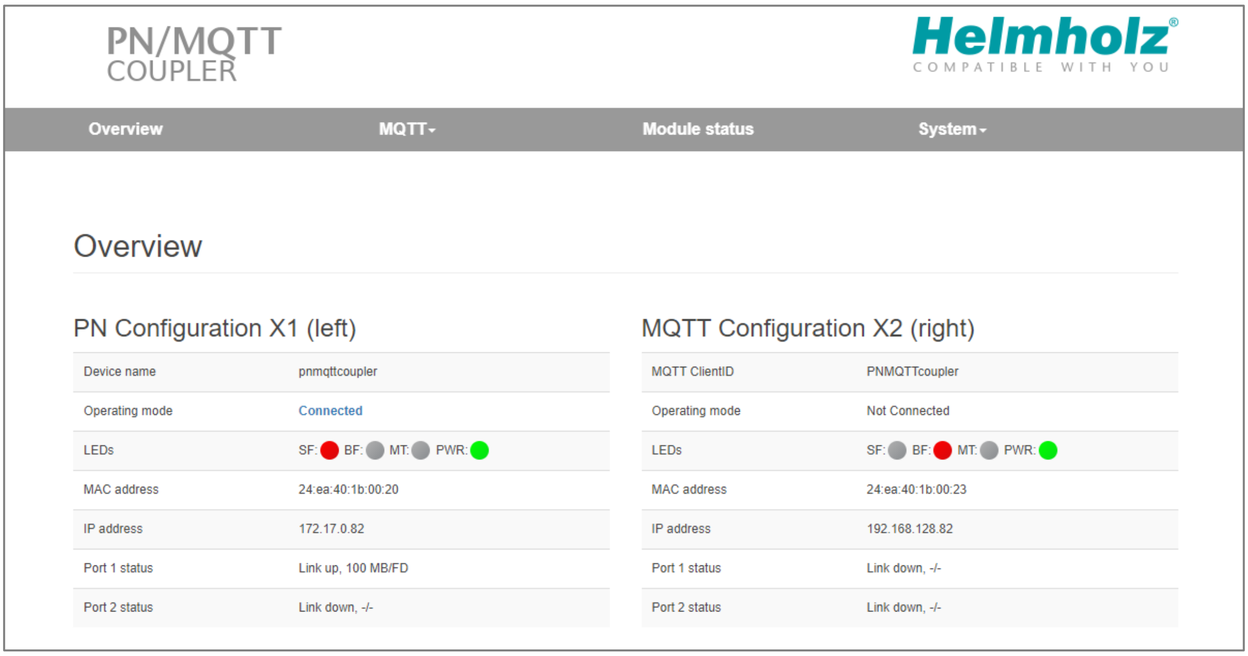
8 Configuration of MQTT connection

8.1 Access to the web page

As soon as the PN/MQTT coupler has been configured via the PROFINET PLC, the web page of the device is accessible via the PROFINET network. If the IP address is also available on the MQTT network page (static IP, DHCP successful), the web page is also accessible via the MQTT network.

NOTE Please note in the commissioning phase in the PROFINET configuration to activate at least the web page on the PROFINET network side (X1).

When accessing the device for the first time, a password must be assigned for the user "admin" with at least 8 characters. After logging in, you will see the "Overview" view:



Overview

PN Configuration X1 (left)

Device name	pnmqttcoupler
Operating mode	Connected
LEDs	SF:🔴 BF:⚫ MT:⚫ PWR:🟢
MAC address	24:ea:40:1b:00:20
IP address	172.17.0.82
Port 1 status	Link up, 100 MB/FD
Port 2 status	Link down, -/-

MQTT Configuration X2 (right)

MQTT ClientID	PNMQTTcoupler
Operating mode	Not Connected
LEDs	SF:⚫ BF:🔴 MT:⚫ PWR:🟢
MAC address	24:ea:40:1b:00:23
IP address	192.168.128.82
Port 1 status	Link down, -/-
Port 2 status	Link down, -/-

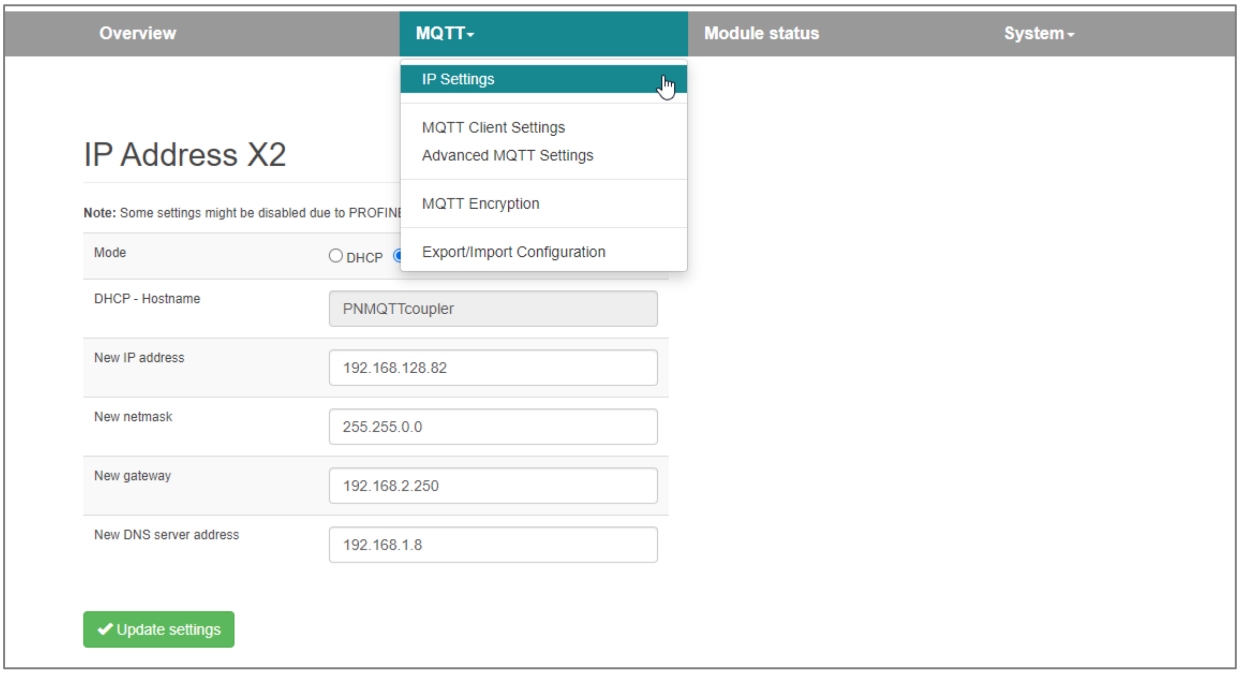
The PN/MQTT Coupler still shows "Bus Error (BF)" on the "Overview" page on the X2 interface in this state because no connection to the MQTT broker has been configured yet.

The error "System error (SF)" on the PROFINET side is also displayed due to the not configured MQTT connection.

NOTE If the web page of the device is not accessible, please check the parameter "Web page" in the PROFINET configuration (see Chap. 7.1), as well as the correct specification of the IP address and subnet mask matching the device with which you call the web page.
Please note that the website has an inactivity timeout. If you do not access the website for a while, you will be logged out.

8.2 MQTT basic configuration

After the configuration of the PROFINET side, the connection to the MQTT broker must still be configured on the web page of the PN/MQTT coupler. The configuration can be done in the "**MQTT**" menu. First select the "**IP Settings**".

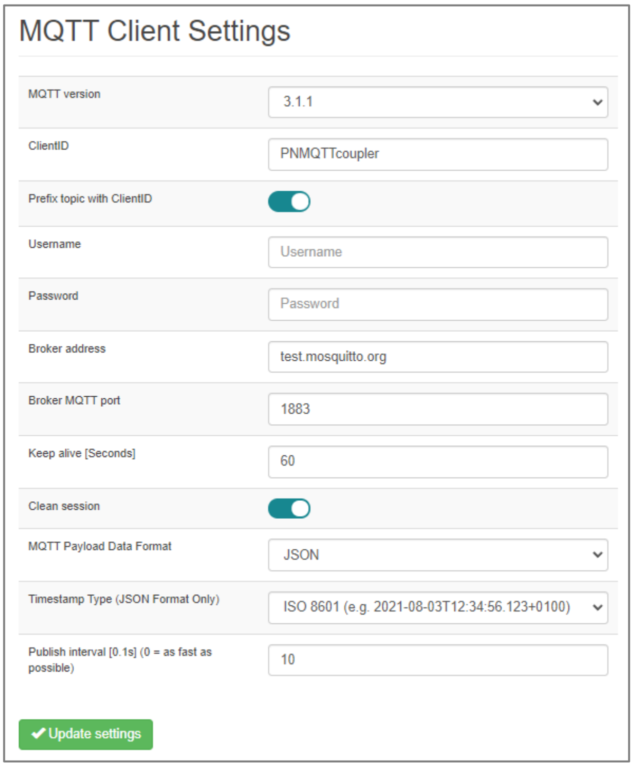


IP Address X2

Note: Some settings might be disabled due to PROFINI

Mode	○ DHCP ●
DHCP - Hostname	PNMQTTcoupler
New IP address	192.168.128.82
New netmask	255.255.0.0
New gateway	192.168.2.250
New DNS server address	192.168.1.8

In the section "IP Address X2" the IP address of the right network port "X2" of the PN/MQTT Coupler is displayed. This can be set if it has not already been received via the PROFINET configuration "Static IP" or via "DHCP".



MQTT Client Settings

MQTT version	3.1.1
ClientID	PNMQTTcoupler
Prefix topic with ClientID	●
Username	Username
Password	Password
Broker address	test.mosquitto.org
Broker MQTT port	1883
Keep alive [Seconds]	60
Clean session	●
MQTT Payload Data Format	JSON
Timestamp Type (JSON Format Only)	ISO 8601 (e.g. 2021-08-03T12:34:56.123+0100)
Publish interval [0.1s] (0 = as fast as possible)	10

The MQTT broker is addressed via interface X2. If the MQTT broker is located in the same network as the PROFINET PLC, see note in chapter 5.2.

The necessary information for the connection with the MQTT broker can be made in the "**MQTT**" menu under "**MQTT Client Settings**".

MQTT version: The PN/MQTT Coupler supports the MQTT standard "3.1.1" and the new standard "V5". Since the two standards are not compatible, the MQTT version must be set to match the broker.

ClientID: Name of the MQTT client when logging on to a broker

Prefix topic with ClientID: With this option, the ClientID of the device can be preassigned to each topic. The topic name "temperature" then becomes "<ClientID>/temperature".

Username/Password: Authentication at the broker

Broker address: IP address or domain name of the broker. The broker must be in the same subnet as the IP address of the coupler's X2 network.

Broker TCP Port: Port for the MQTT connection to the broker. Common are "1883" for unencrypted and "8883" for TLS encrypted connections.

Keep alive: Time for sending the cyclic life message of the coupler to the broker. If this message is missing, the broker assumes a failure of the coupler.

Clean session (MQTT V3.1.1): Information to the broker upon establishing the connection whether old messages should be deleted or reused.

Clean start (MQTT V5): If Clean Start is enabled, the client and broker must discard existing sessions and start a new session when a connection is established.
If Clean Start is disabled and a session is associated with this client ID, the broker must resume communication with the client based on the status of the session. If no session is associated with this client ID, the broker must create a new session.

Session expiry interval [Seconds] (MQTT 5.0 only): In the context of "Clean start", if "Session expiry interval" is set to 0, the session is terminated when the network connection is closed. Otherwise, the session is kept open until the time expires.

MQTT payload data format:
The value of a topic can be sent in simple text form ("Text") or in structured form ("JSON). More information about the representation of the values is explained in chapter 8.3.

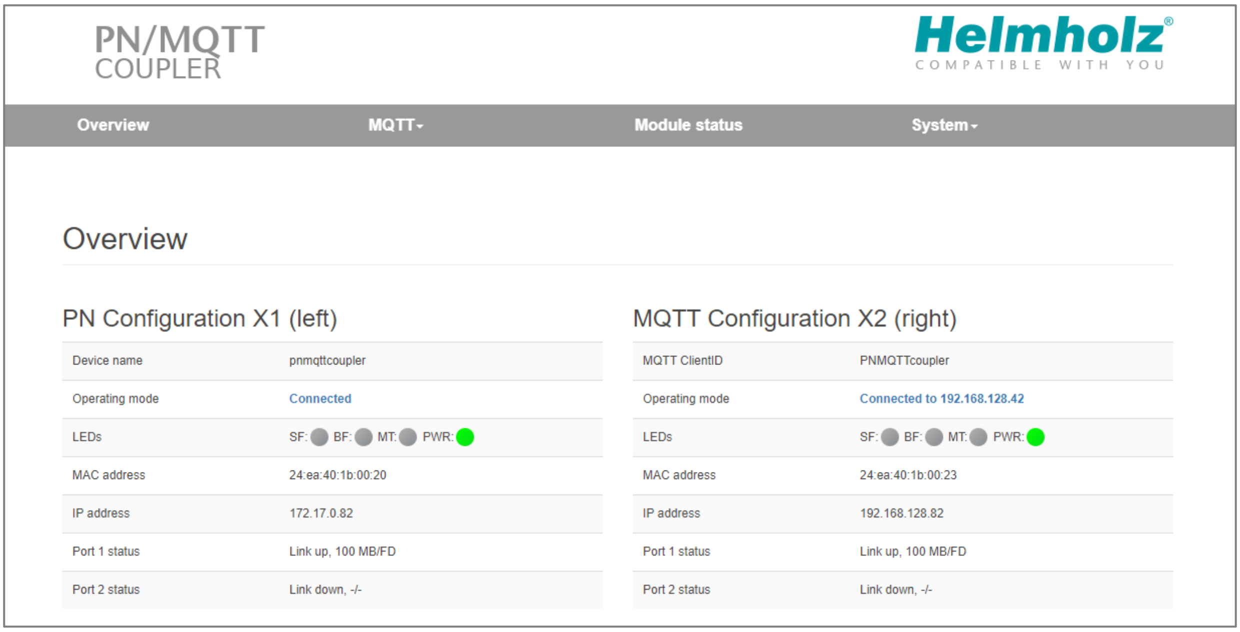
Timestamp Type: The PN/MQTT coupler can (only for JSON formatted messages) add a timestamp to the message.

Publish interval: An MQTT message for a slot is sent automatically as soon as the value changes. If PLC values are to be sent that change very quickly, it is possible to limit the send interval. The Publish interval '0' tells the coupler to send as fast as possible. A number greater than zero tells the coupler not to send faster than x * 0.1 seconds.

8.3 Establish and check MQTT Broker connection

If all basic settings have been set correctly and accepted with "Update settings", the PN/MQTT Coupler should automatically connect to the MQTT broker and the red LEDs should no longer be displayed.

The "Overview" view can be used to check the state:



Overview

PN Configuration X1 (left)

Device name	pnmqttcoupler
Operating mode	Connected
LEDs	SF: ● BF: ● MT: ● PWR: ●
MAC address	24:ea:40:1b:00:20
IP address	172.17.0.82
Port 1 status	Link up, 100 MB/FD
Port 2 status	Link down, -/-

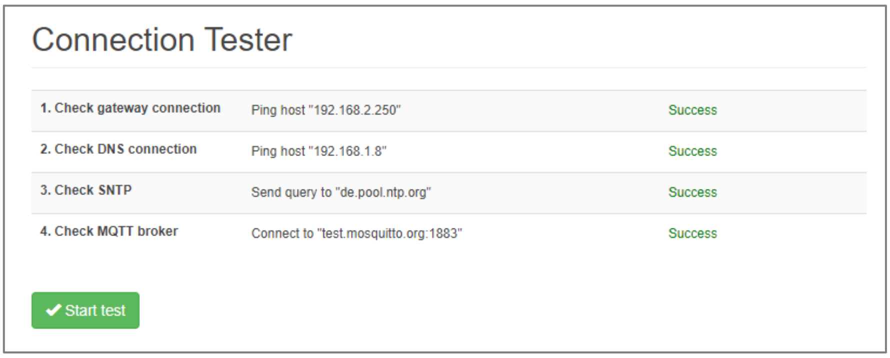
MQTT Configuration X2 (right)

MQTT ClientID	PNMQTTcoupler
Operating mode	Connected to 192.168.128.42
LEDs	SF: ● BF: ● MT: ● PWR: ●
MAC address	24:ea:40:1b:00:23
IP address	192.168.128.82
Port 1 status	Link up, 100 MB/FD
Port 2 status	Link down, -/-

In the next step you can address the I/O data in the PLC program.

To test the MQTT Broker connection, the PN/MQTT Coupler provides a "Connection Tester" in the "MQTT" menu.

The Connection Tester tests in 4 subsequent steps if a connection to the internet can be established if the name resolution and the time server works and if the MQTT Broker port is reachable.



Connection Tester

1. Check gateway connection	Ping host "192.168.2.250"	Success
2. Check DNS connection	Ping host "192.168.1.8"	Success
3. Check SNTP	Send query to "de.pool.ntp.org"	Success
4. Check MQTT broker	Connect to "test.mosquitto.org:1883"	Success

8.4 MQTT Payload formats

The value of a topic can be sent in simple text form ("Text") or in structured form ("JSON”). The setting can only be set globally for all topics together under "MQTT Client Settings".

Example for text format:

```
-12345
```

Some MQTT applications expect a structured form in JSON format.

Example for JSON format:

```
{
   "value": -12345
}
```

The values are displayed differently depending on the data type:

Type	Size	Format	Presentation
Bit	Bit	Text	„0“/„1“, „off“/„on“, „no“/„yes“, „false“/„true“ (parameterizable, see note)
Byte	1 Byte	Hexadecimal	„0x00“ … „0xFF“
Unsigned short Int	1 Byte	Decimal	„0“ … „255“
Signed short Int	1 Byte	Decimal	„-127“ … „128“
Word	2 Bytes	Hexadecimal	„0x0000“ … „0xFFFF“
Unsigned Int	2 Bytes	Decimal	„0“ … „65536“
Signed Int	2 Bytes	Decimal	„-32767“ … „32787“
Double Word	4 Bytes	Hexadecimal	„0x00000000“ … „0xFFFFFFFF“
Unsigned double Int	4 Bytes	Decimal	„0“ … „4294967295“
Signed double Int	4 Bytes	Decimal	„-2147483648“ … „2147483647“
Real	4 Bytes	Floating point	„-123.456789“ (example)
Long Real	8 Bytes	Floating point	„123456.789999“(example)

ATTENTION Modules with data type "Bit" occupy a whole byte in the PLC, because PROFINET does not support bits only. Only the lowest bit of the transmitted byte is used.
If a topic of type Bit is received (Topic Subscription) all above mentioned formats are interpreted and the upper/lower case is arbitrary.

NOTE If you need a different representation of the MQTT payload for your application, please contact us. The payload variants are constantly being expanded.

9 Status and control via the PLC

9.1 Status of PN/MQTT Coupler

The PN/MQTT Coupler provides a status (4 bytes) via the PROFINET input image:

Byte/Bit	7	6	5	4	3	2	1	0
Input Byte 0	PROFINET configuration OK	0	PS 1 Voltage available	PS 2 Voltage available	0	0	X2 network IP address available	X2 network cable detected
Input Byte 1	0	0	0	0	0	0	0	MQTT Broker connection active
Input Byte 2	Last MQTT error code (MQTT V5) or Connect Return/Reason-Code							
Input Byte 3	*Reserved*							

To check the correct function of the PN/MQTT coupler in the PLC, the bit "PROFINET configuration OK" and "MQTT broker connection active" should be read.

9.2 Control of PN/MQTT Coupler

The PN/MQTT coupler can be controlled via the following control bits (1 byte) in the PROFINET output image:

Byte/Bit	7	6	5	4	3	2	1	0
Output Byte 0	Clear MQTT Error Code		-	-	-	-	MQTT Disconnect connection	MQTT lock data exchange

9.3 Subscribe Module

In addition to the input data for the actual value, the subscriber modules additionally have a status byte and a control byte.

Status bits of Subscribe Module:

Byte/Bit	7	6	5	4	3	2	1	0
Input Byte 0	1 = new data received	Receive counter						

Control bits of Subscribe Module:

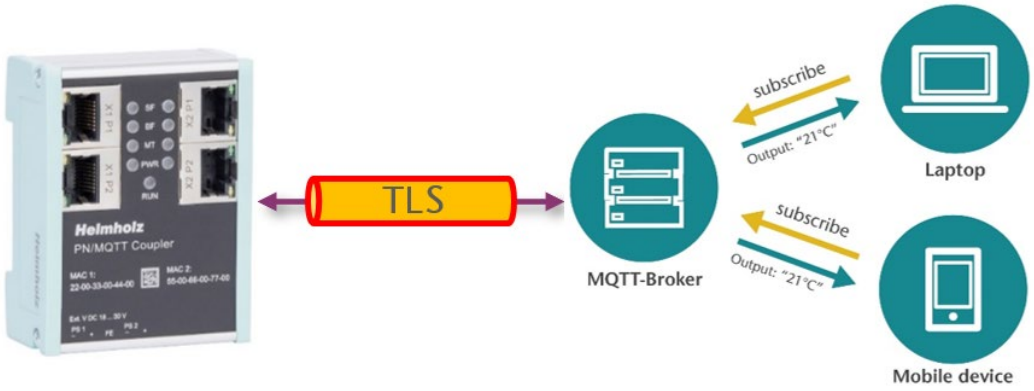
Byte/Bit	7	6	5	4	3	2	1	0
Output Byte 0	1 = Reset data receive bit	-	-	-	-	-	-	-

To be able to detect the reception of MQTT messages, the receive bit 7 can be used, which must always be reset in the output byte of the module. Alternatively, the receive counter can be checked for changes.

Attention: the receive counter runs until 0x7F and is then reset to 0x00.

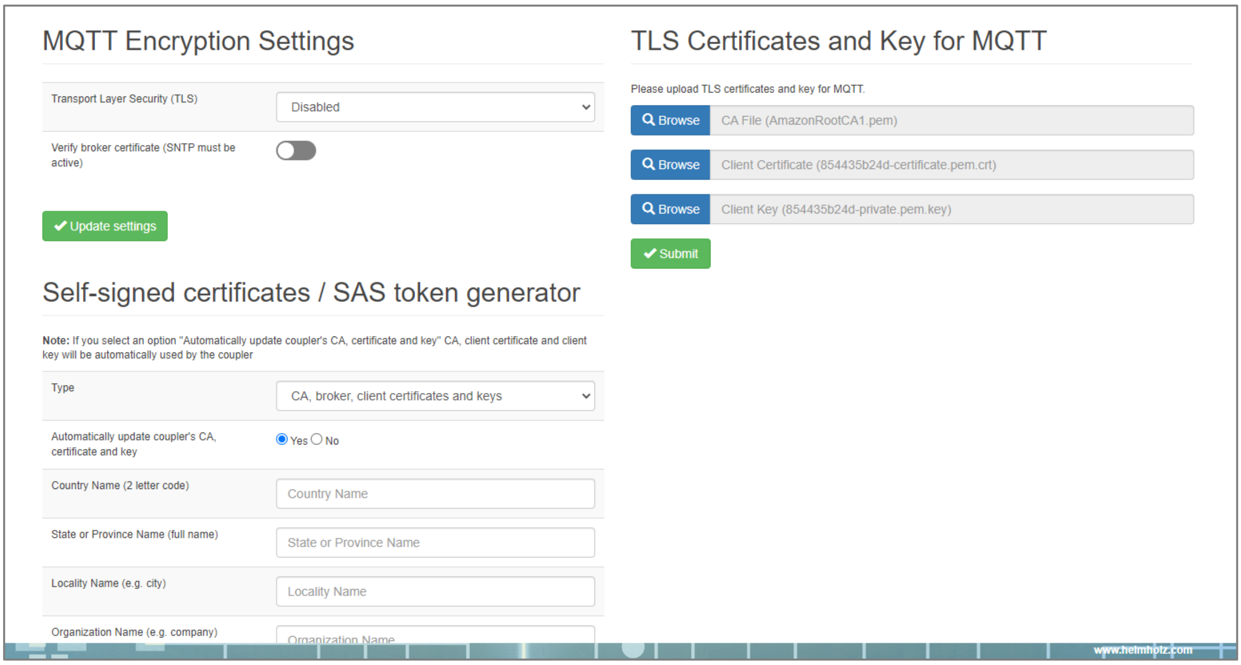
10 MQTT encryption and authentication

The transmission between the client and the broker can be encrypted and the devices can authenticate each other, i.e. verify their identity.



Encryption prevents a third party from reading the data. Authentication ensures that only the right devices can exchange data with each other.

In the "MQTT" menu under "MQTT Encryption", encryption can be activated, certificates can be stored, and self-generated certificates can also be created.



MQTT Encryption Settings

Transport Layer Security (TLS)	Disabled
Verify broker certificate (SNTP must be active)	○

✔ Update settings

Self-signed certificates / SAS token generator

Note: If you select an option "Automatically update coupler's CA, certificate and key" CA, client certificate and client key will be automatically used by the coupler

Type	CA, broker, client certificates and keys
Automatically update coupler's CA, certificate and key	● Yes ○ No
Country Name (2 letter code)	Country Name
State or Province Name (full name)	State or Province Name
Locality Name (e.g. city)	Locality Name
Organization Name (e.g. company)	Organization Name

TLS Certificates and Key for MQTT

Please upload TLS certificates and key for MQTT.

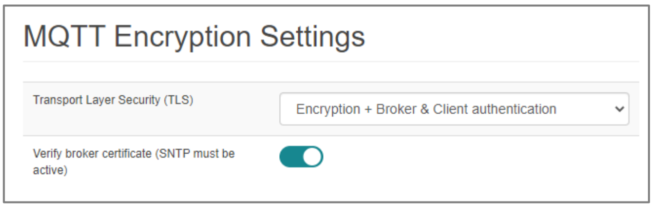
Browse CA File (AmazonRootCA1.pem)

Browse Client Certificate (854435b24d-certificate.pem.crt)

Browse Client Key (854435b24d-private.pem.key)

✔ Submit

Transport Layer Security (TLS):
Disabled: unencrypted data exchange between client and broker. No certificates or keys are required.



MQTT Encryption Settings

Transport Layer Security (TLS)	Encryption + Broker & Client authentication
Verify broker certificate (SNTP must be active)	●

Enabled – Encryption only: Enables encryption without authentication. This option does not require a CA, client certificate or client key.

Encryption + Broker authentication: enables encryption with broker authentication done by the client. With this option, a CA must be uploaded for broker verification (see below).

Encryption + Broker & Client authentication: enables encryption with mutual broker and client authentication. This option requires a CA and client certificate. In addition to broker verification by the client, the broker can also verify the client as it sends its certificate during the TLS handshake.

Verify broker certificate: Certificates contain an expiration date and must be updated regularly. This option checks whether the broker certificate is still valid.

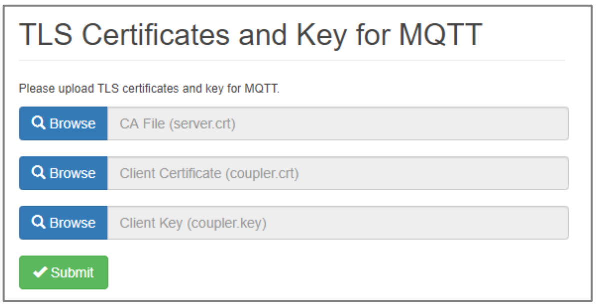
ATTENTION When using certificates for authentication, time synchronization of the PN/MQTT coupler using SNTP must be enabled.

For encryption and authentication, certificates and keys must be provided (uploaded) to the PN/MQTT Coupler.

CA File: certificate of broker

Client Certificate: Certificate for the PN/MQTT Coupler

Client Key: Private Key File for the PN/MQTT Coupler



TLS Certificates and Key for MQTT

Please upload TLS certificates and key for MQTT.

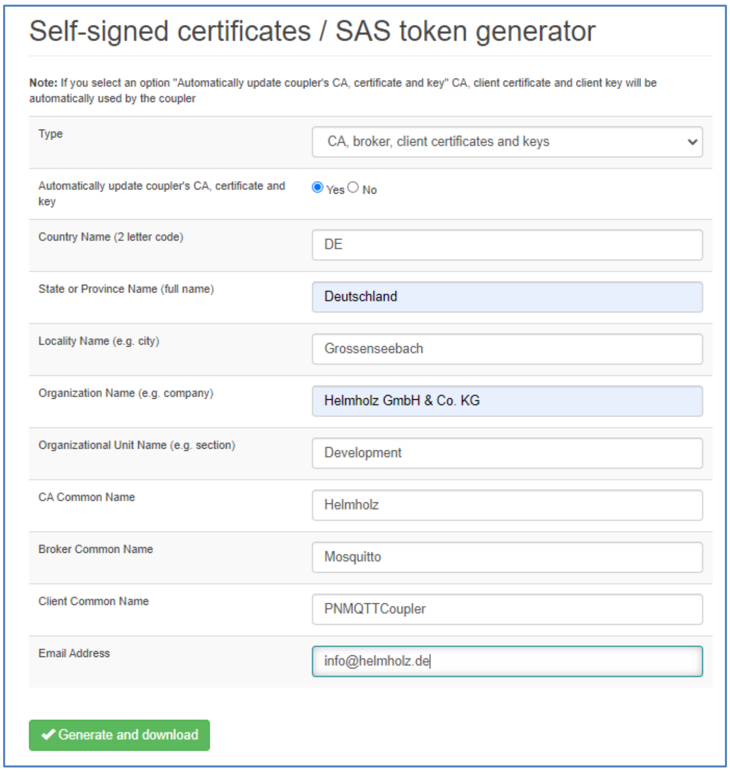
10.1 Generator for certificates and SAS tokens

For an encrypted and authenticated connection with a broker - whether "on premise" or in the cloud - the certificate of the broker and the certificate of the parent certification authority ('CA') should be downloadable or generated by IT for the own network.

The certificate for the client should then also either be generated by the broker application (example "Amazon IoT", see Chapter 17) or also created by IT.

To make it easier to work with certificates during internal tests, the PN/MQTT Coupler includes a built-in generator for self-signed certificates. For use with Microsoft Azure, a SAS token generator is also available (see Chapter 18).

If an encrypted and authenticated connection is to be established for a test setup with a local broker (e.g. Mosquitto), the PN/MQTT coupler can create the certificates and the private keys both for the PN/MQTT coupler itself and for the broker with the option "**CA, broker, client certificates and keys**". The certificate of the local certificate authority ('CA') with which the other two certificates were signed is also supplied.



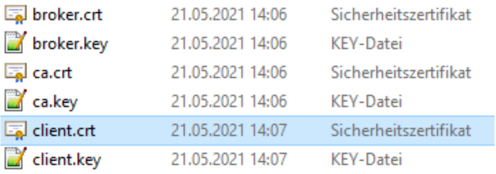
Self-signed certificates / SAS token generator

Note: If you select an option "Automatically update coupler's CA, certificate and key" CA, client certificate and client key will be automatically used by the coupler

Type	CA, broker, client certificates and keys
Automatically update coupler's CA, certificate and key	● Yes ○ No
Country Name (2 letter code)	DE
State or Province Name (full name)	Deutschland
Locality Name (e.g. city)	Grossenseebach
Organization Name (e.g. company)	Helmholz GmbH & Co. KG
Organizational Unit Name (e.g. section)	Development
CA Common Name	Helmholz
Broker Common Name	Mosquitto
Client Common Name	PNMQTTCoupler
Email Address	info@helmholz.de

The content of the input fields in the dialog are transferred to the certificates, but they have a more informative character.

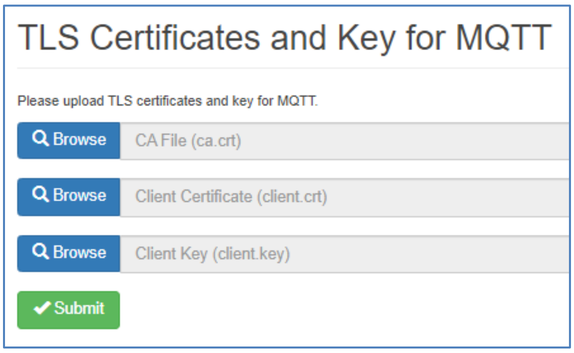
After pressing the "Generate and Download" button, the certificates are generated and a ZIP file with the certificates is downloaded:



broker.crt	21.05.2021 14:06	Sicherheitszertifikat
broker.key	21.05.2021 14:06	KEY-Datei
ca.crt	21.05.2021 14:06	Sicherheitszertifikat
ca.key	21.05.2021 14:06	KEY-Datei
client.crt	21.05.2021 14:07	Sicherheitszertifikat
client.key	21.05.2021 14:07	KEY-Datei

The option "Automatically update coupler's CA, certificate and key" transfers the relevant files directly to the PN/MQTT coupler.

In the broker, its private key, certificate, and CA file can now also be used.

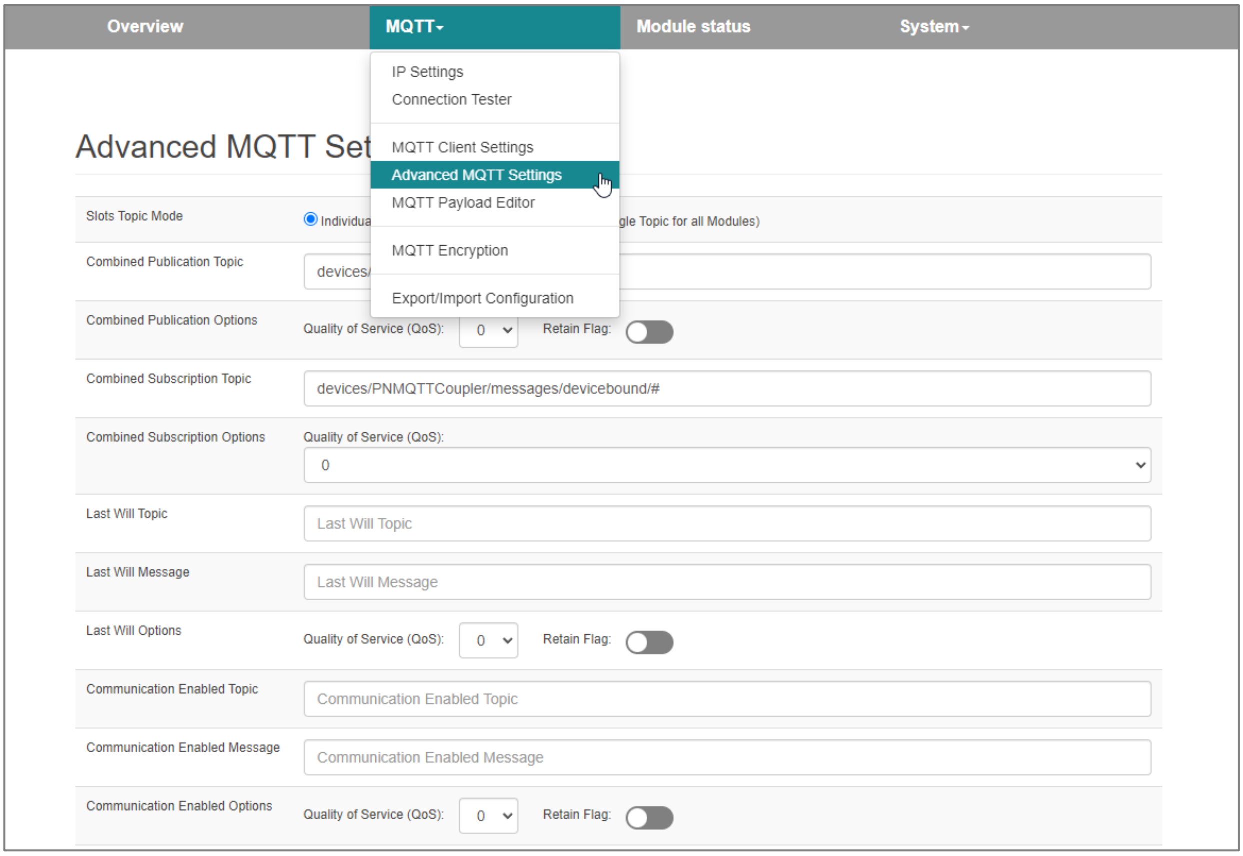


TLS Certificates and Key for MQTT

Please upload TLS certificates and key for MQTT.

11 More MQTT settings

Further settings for the MQTT behavior of the PN/MQTT Coupler can be made using the "Advanced MQTT Settings" dialog in the “MQTT” menu.



Advanced MQTT Set

Slots Topic Mode: ● Individua … gle Topic for all Modules)

Combined Publication Topic: devices/

Combined Publication Options: Quality of Service (QoS): 0 Retain Flag: ○

Combined Subscription Topic: devices/PNMQTTCoupler/messages/devicebound/#

Combined Subscription Options: Quality of Service (QoS): 0

Last Will Topic: Last Will Topic

Last Will Message: Last Will Message

Last Will Options: Quality of Service (QoS): 0 Retain Flag: ○

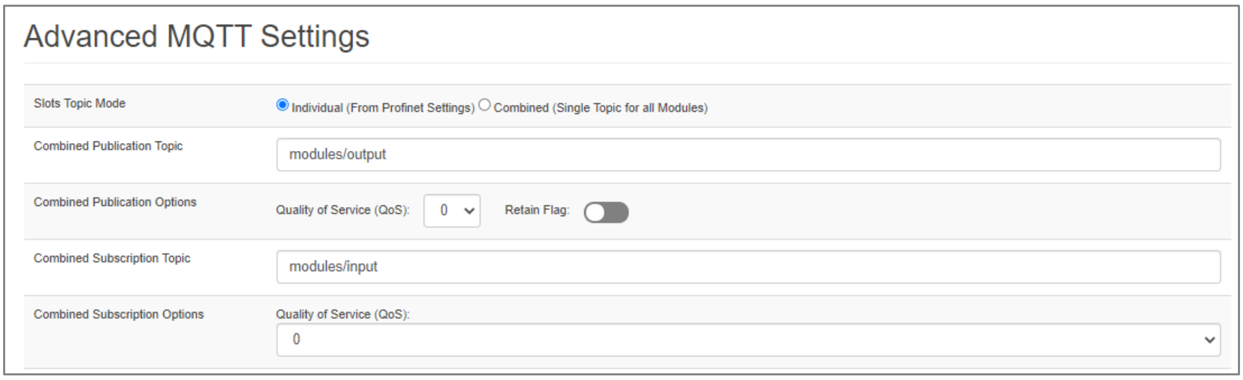
Communication Enabled Topic: Communication Enabled Topic

Communication Enabled Message: Communication Enabled Message

Communication Enabled Options: Quality of Service (QoS): 0 Retain Flag: ○

11.1 Topic Mode

The topic mode determines how all configured topic messages are sent. Usually, each topic (each configured module) is sent or received individually as an MQTT message Topic Mode *"Individual"*.



Advanced MQTT Settings

Slots Topic Mode: ● Individual (From Profinet Settings) ○ Combined (Single Topic for all Modules)

Combined Publication Topic: modules/output

Combined Publication Options: Quality of Service (QoS): 0 Retain Flag: ○

Combined Subscription Topic: modules/input

Combined Subscription Options: Quality of Service (QoS): 0

For certain applications, e.g. to connect to the Microsoft Azure Cloud (see Chapter 18), a device may only send or receive all data under one MQTT message Topic Mode "Combined".

For the Combined use case, the Topic Name for Publishing and the Topic Name for Subscription can be specified in the following settings, along with the associated QoS and Retain property.

Example of Topic Mode "Individual":

Message 1 for Topic „temperature“

```
{
   "value": 23
}
```

Message 2 for Topic „humidity“:

```
{
   "value": 40
}
```

Example for Topic Mode "Combined":

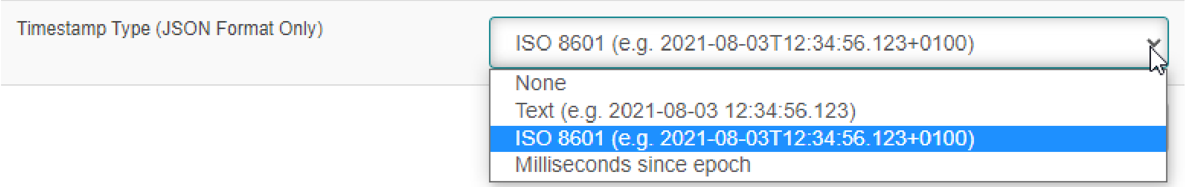
Message for Topic „modules/output“

```
{
   "temperature": 23
   "humidity": 40
}
```

11.2 Timestamp for Topic messages (only in JSON)

In many applications it is important to get a time information with the data information in the message to be able to store the time belonging to the date.

In the MQTT Client Settings menu it is possible - exclusively for JSON messages - to set that a timestamp is sent with the message.

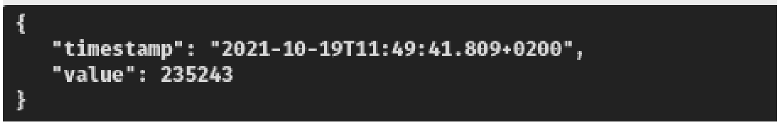


Timestamp Type (JSON Format Only): ISO 8601 (e.g. 2021-08-03T12:34:56.123+0100)

None
Text (e.g. 2021-08-03 12:34:56.123)
ISO 8601 (e.g. 2021-08-03T12:34:56.123+0100)
Milliseconds since epoch

There are 3 different timestamp formats available: Text, ISO8601 and Milliseconds since epoch.

Example with ISO 8601 time stamp:



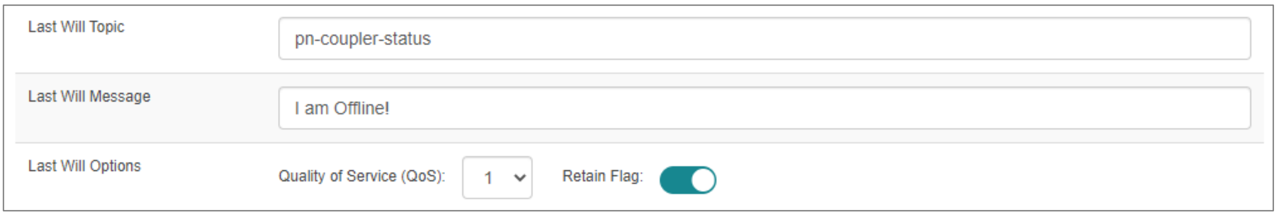
```
{
   "timestamp": "2021-10-19T11:49:41.809+0200",
   "value": 235243
}
```

NOTE

For more complex MQTT messages, the MQTT Payload Editor can be used. See chapter 11.5.

11.3 Last Will Message

The "Last Will Message" is an MQTT message to inform other clients about an improperly disconnected client. The PN/MQTT coupler sends its "last will" message to the broker when it connects to the broker. Under the menu "Advanced MQTT Settings" the Last Will Message can be set.



Last Will Topic	pn-coupler-status
Last Will Message	I am Offline!
Last Will Options	Quality of Service (QoS): 1 Retain Flag: ●

The "last will" message is a normal MQTT message with an arbitrary topic and payload. The broker stores the message until it detects that the client has involuntarily disconnected. In response to the unforeseen disconnection, the broker sends the "last-will" message to all subscribed clients of the "last-will" topic.

If the client disconnects properly, the broker discards the stored "Last Will Message".

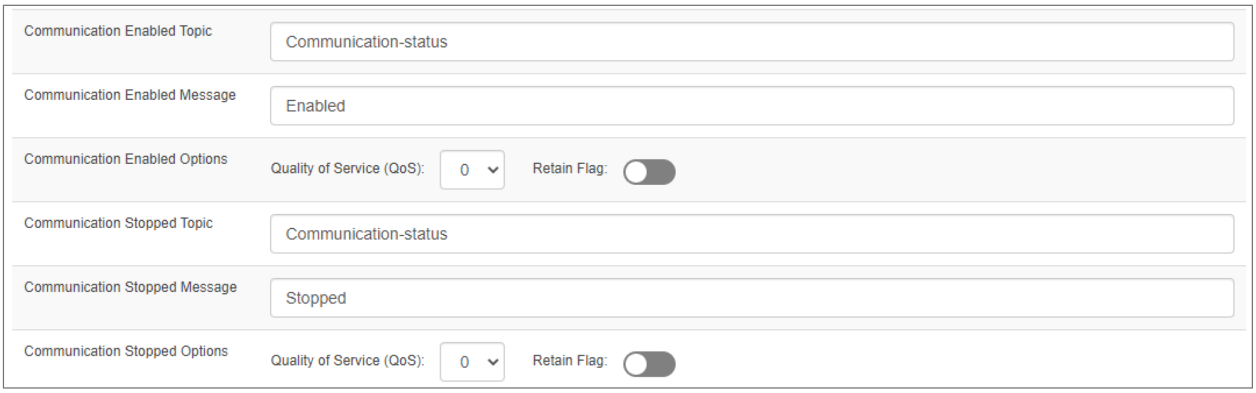
NOTE

Not all brokers support the Last Will Message.

11.4 „Communication Enable“ and „Communication Stopped“ messages

The "Communication Enable" message is always sent by the PN/MQTT coupler when the coupler is ready for operation. For this the coupler must be configured via PROFINET and the PLC must be in RUN.

The "Communication Stopped" message is always sent by the PN/MQTT coupler when the coupler is no longer ready for operation. Reasons for this are a network interruption, a reconfiguration of the device or if the PLC has stopped.

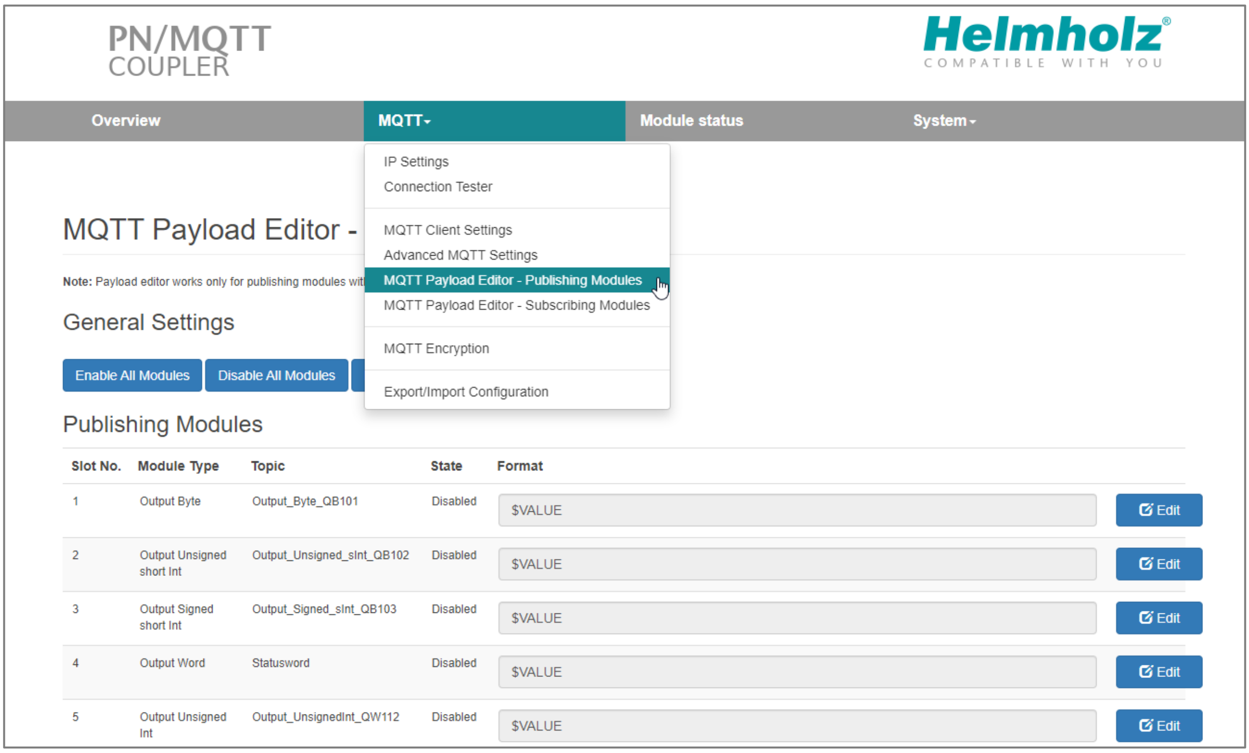


Communication Enabled Topic	Communication-status
Communication Enabled Message	Enabled
Communication Enabled Options	Quality of Service (QoS): 0 Retain Flag: ○
Communication Stopped Topic	Communication-status
Communication Stopped Message	Stopped
Communication Stopped Options	Quality of Service (QoS): 0 Retain Flag: ○

11.5 Payload Editor

The content of an MQTT message may require a complex structure depending on the application. The PN/MQTT Coupler provides a payload editor for this purpose.

For each Publish Topic the format of the message can be defined separately and independently with the "MQTT Payload Editor".



MQTT Payload Editor -

Note: Payload editor works only for publishing modules wit

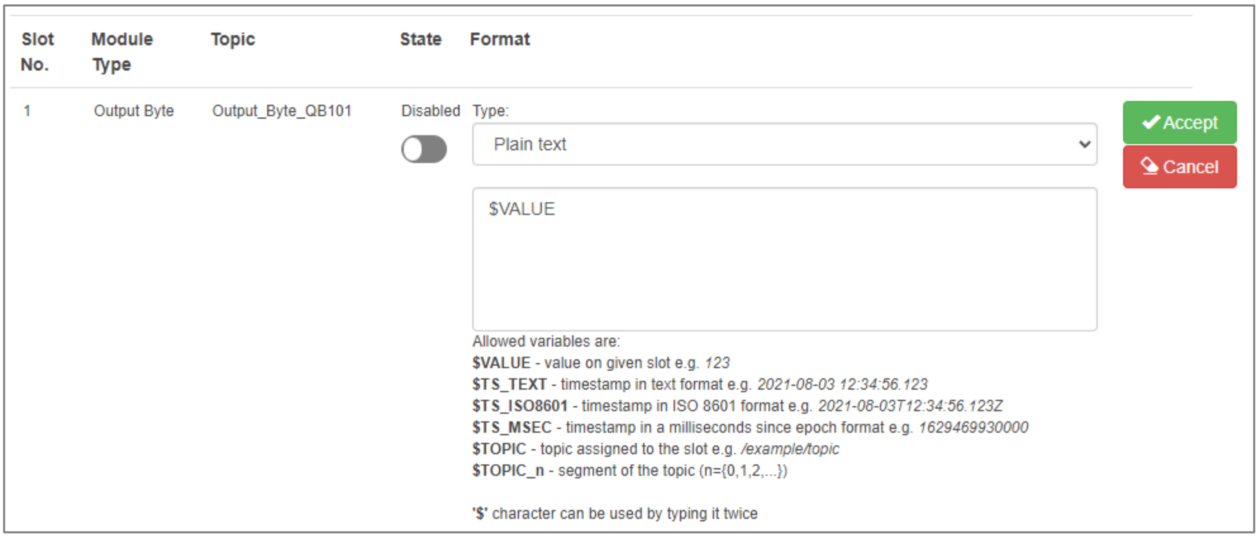
General Settings

Publishing Modules

Slot No.	Module Type	Topic	State	Format
1	Output Byte	Output_Byte_QB101	Disabled	$VALUE
2	Output Unsigned short Int	Output_Unsigned_sInt_QB102	Disabled	$VALUE
3	Output Signed short Int	Output_Signed_sInt_QB103	Disabled	$VALUE
4	Output Word	Statusword	Disabled	$VALUE
5	Output Unsigned Int	Output_UnsignedInt_QW112	Disabled	$VALUE

In the default state, the payload editor is switched off for all topics (state = "Disabled"). In this state, the payload of the topic is sent as set in the "MQTT Client Settings".

To set an individual payload format for a topic, the "Edit" button must be pressed:



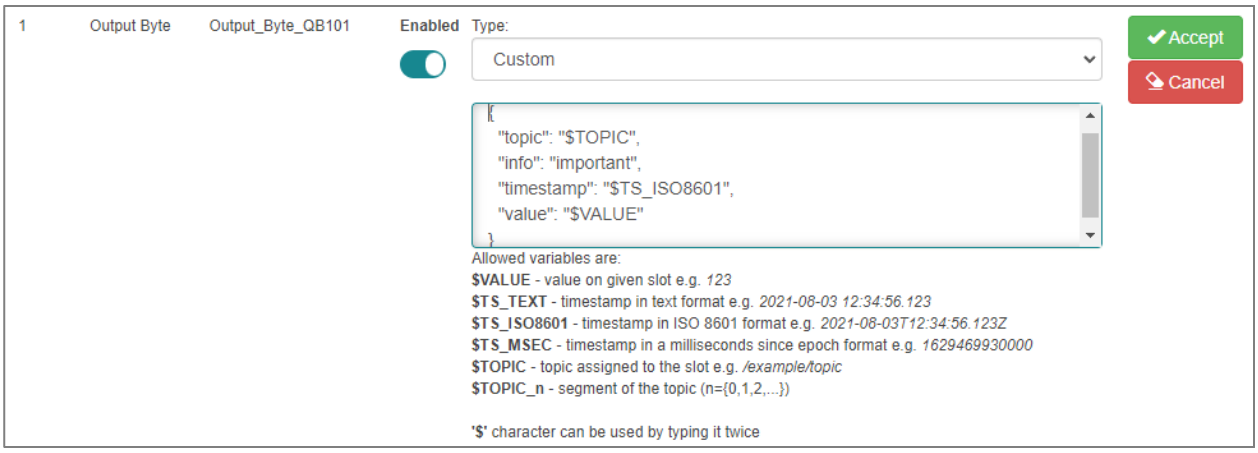
Slot No.	Module Type	Topic	State	Format
1	Output Byte	Output_Byte_QB101	Disabled	Type: Plain text $VALUE

Allowed variables are:
$VALUE - value on given slot e.g. *123*
$TS_TEXT - timestamp in text format e.g. *2021-08-03 12:34:56.123*
$TS_ISO8601 - timestamp in ISO 8601 format e.g. *2021-08-03T12:34:56.123Z*
$TS_MSEC - timestamp in a milliseconds since epoch format e.g. *1629469930000*
$TOPIC - topic assigned to the slot e.g. */example/topic*
$TOPIC_n - segment of the topic (n={0,1,2,...})

'**$**' character can be used by typing it twice

With "Type" you can select whether the payload should be formatted as "Plain Text", "JSON" or "Custom" (freely editable). With the setting "Custom" the message can be built up freely in an editor in the dialog-field under “Type”.

Here is an example of a freely structured JSON message:



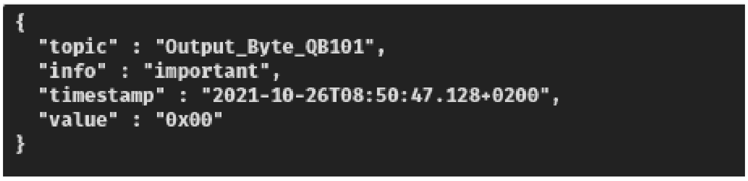
```
{
  "topic": "$TOPIC",
  "info": "important",
  "timestamp": "$TS_ISO8601",
  "value": "$VALUE"
}
```

Allowed variables are:
$VALUE - value on given slot e.g. *123*
$TS_TEXT - timestamp in text format e.g. *2021-08-03 12:34:56.123*
$TS_ISO8601 - timestamp in ISO 8601 format e.g. *2021-08-03T12:34:56.123Z*
$TS_MSEC - timestamp in a milliseconds since epoch format e.g. *1629469930000*
$TOPIC - topic assigned to the slot e.g. */example/topic*
$TOPIC_n - segment of the topic (n={0,1,2,...})

'**$**' character can be used by typing it twice

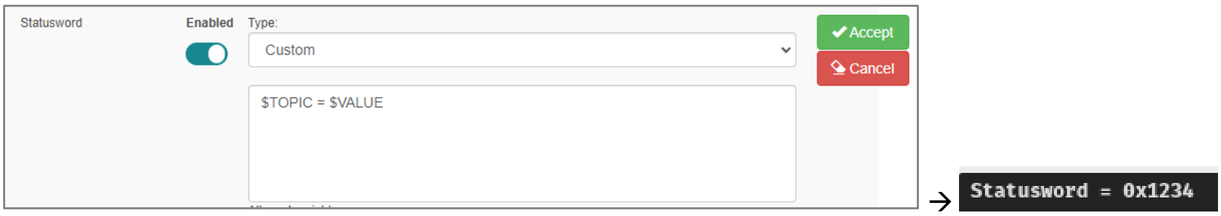
With the switch to the left of "Type" the message format set in the Payload Editor can be activated ("Enabled") or ignored ("Disabled").

Here is an example transmission of the topic "Output_Byte_QB101":



```
{
  "topic" : "Output_Byte_QB101",
  "info" : "important",
  "timestamp" : "2021-10-26T08:50:47.128+0200",
  "value" : "0x00"
}
```

The Payload Editor can also be used to send arbitrarily formatted messages. An example:



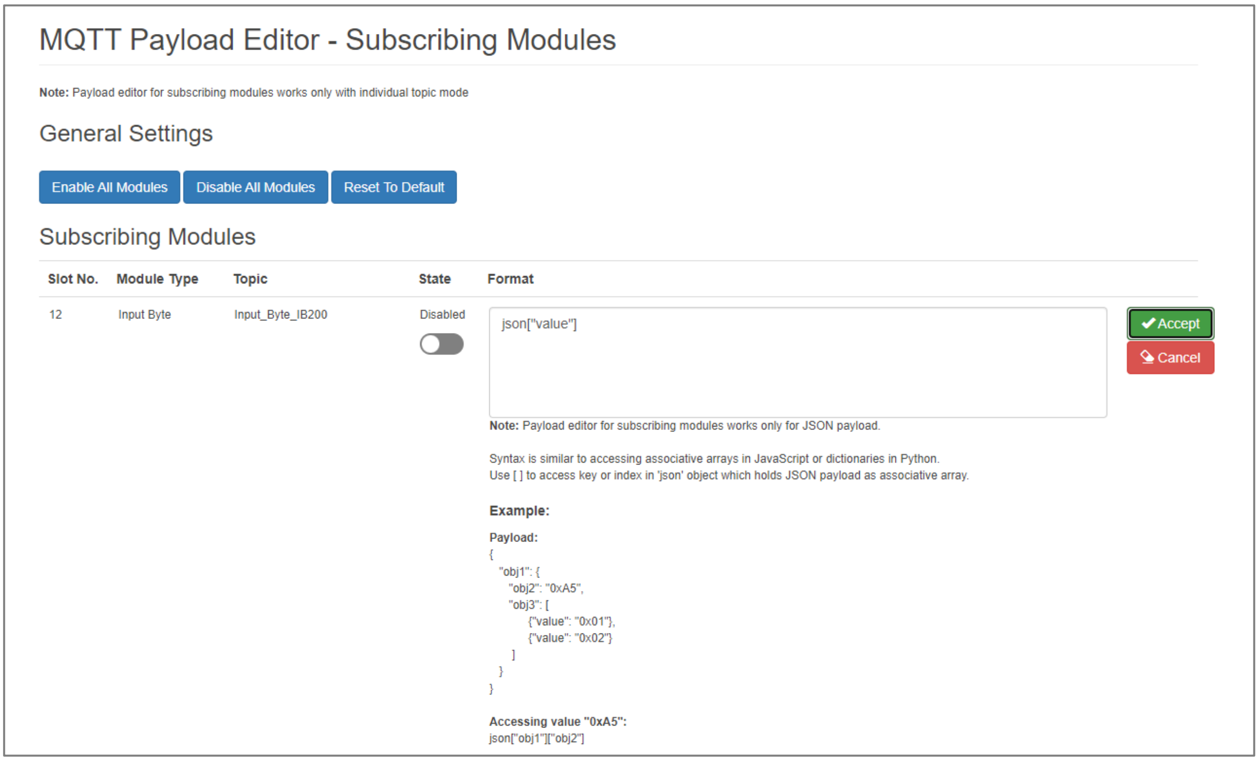
```
$TOPIC = $VALUE
```

→

```
Statusword = 0x1234
```

NOTE If you need a specific representation of the MQTT payload for your application, please contact us. We are happy to support you.

The Payload Editor for receiving messages (Subscribing Modules) works according with a similar concept. However, the coding in the payload editor now have the function of extracting the desired value from a complex message. The payload editor necessarily assumes that the received message is JSON formatted. In the received JSON structure, the value can now be selected via the named object within the structure.



MQTT Payload Editor - Subscribing Modules

Note: Payload editor for subscribing modules works only with individual topic mode

General Settings

Subscribing Modules

Slot No.	Module Type	Topic	State	Format
12	Input Byte	Input_Byte_IB200	Disabled	json["value"]

Note: Payload editor for subscribing modules works only for JSON payload.

Syntax is similar to accessing associative arrays in JavaScript or dictionaries in Python.
Use [ ] to access key or index in 'json' object which holds JSON payload as associative array.

Example:

Payload:

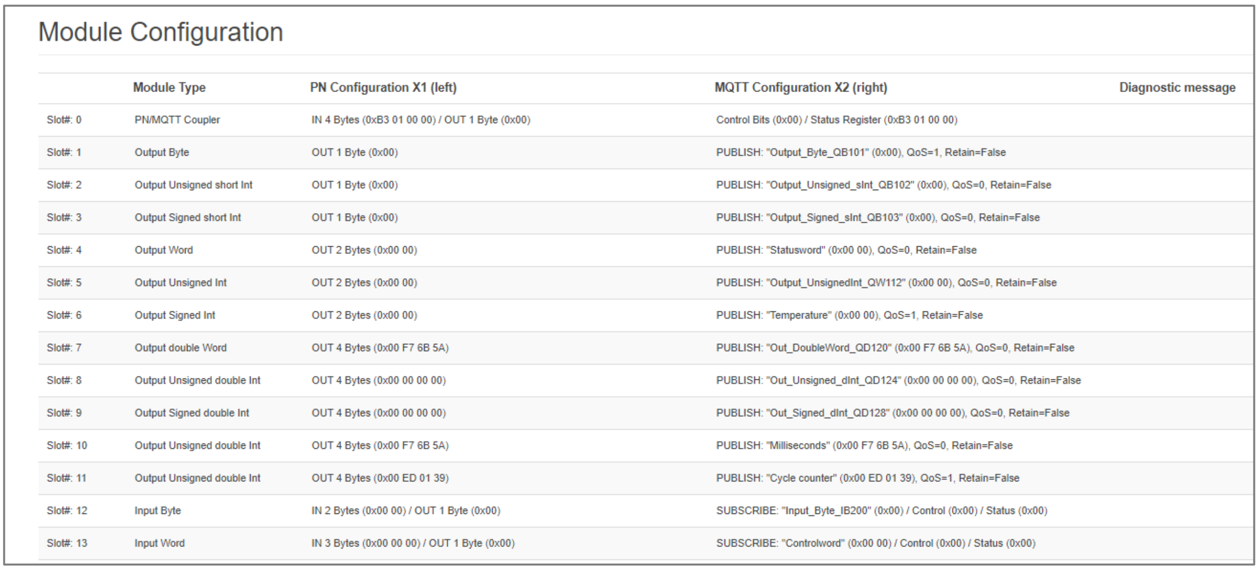
```
{
  "obj1": {
    "obj2": "0xA5",
    "obj3": [
        {"value": "0x01"},
        {"value": "0x02"}
     ]
  }
}
```

Accessing value "0xA5":
json["obj1"]["obj2"]

12 Further functions in the web interface

12.1 Module status

On the web page "Module status" the configured module configuration and the current IO data are displayed. If there is a configuration error, the error is displayed in the "Diagnostic message" column.



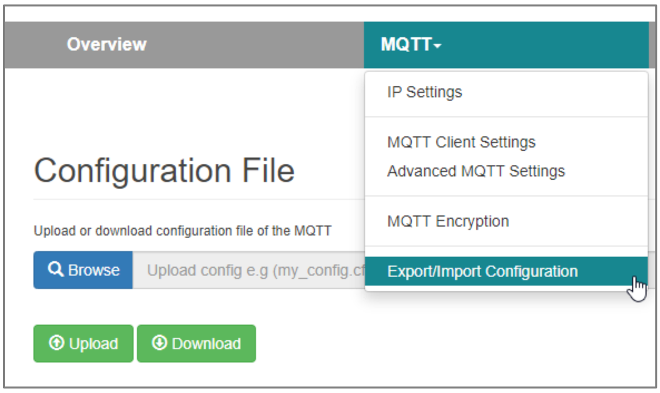
Module Configuration

	Module Type	PN Configuration X1 (left)	MQTT Configuration X2 (right)	Diagnostic message
Slot#: 0	PN/MQTT Coupler	IN 4 Bytes (0xB3 01 00 00) / OUT 1 Byte (0x00)	Control Bits (0x00) / Status Register (0xB3 01 00 00)	
Slot#: 1	Output Byte	OUT 1 Byte (0x00)	PUBLISH: "Output_Byte_QB101" (0x00), QoS=1, Retain=False	
Slot#: 2	Output Unsigned short Int	OUT 1 Byte (0x00)	PUBLISH: "Output_Unsigned_sInt_QB102" (0x00), QoS=0, Retain=False	
Slot#: 3	Output Signed short Int	OUT 1 Byte (0x00)	PUBLISH: "Output_Signed_sInt_QB103" (0x00), QoS=0, Retain=False	
Slot#: 4	Output Word	OUT 2 Bytes (0x00 00)	PUBLISH: "Statusword" (0x00 00), QoS=0, Retain=False	
Slot#: 5	Output Unsigned Int	OUT 2 Bytes (0x00 00)	PUBLISH: "Output_UnsignedInt_QW112" (0x00 00), QoS=0, Retain=False	
Slot#: 6	Output Signed Int	OUT 2 Bytes (0x00 00)	PUBLISH: "Temperature" (0x00 00), QoS=1, Retain=False	
Slot#: 7	Output double Word	OUT 4 Bytes (0x00 F7 6B 5A)	PUBLISH: "Out_DoubleWord_QD120" (0x00 F7 6B 5A), QoS=0, Retain=False	
Slot#: 8	Output Unsigned double Int	OUT 4 Bytes (0x00 00 00 00)	PUBLISH: "Out_Unsigned_dInt_QD124" (0x00 00 00 00), QoS=0, Retain=False	
Slot#: 9	Output Signed double Int	OUT 4 Bytes (0x00 00 00 00)	PUBLISH: "Out_Signed_dInt_QD128" (0x00 00 00 00), QoS=0, Retain=False	
Slot#: 10	Output Unsigned double Int	OUT 4 Bytes (0x00 F7 6B 5A)	PUBLISH: "Milliseconds" (0x00 F7 6B 5A), QoS=0, Retain=False	
Slot#: 11	Output Unsigned double Int	OUT 4 Bytes (0x00 ED 01 39)	PUBLISH: "Cycle counter" (0x00 ED 01 39), QoS=1, Retain=False	
Slot#: 12	Input Byte	IN 2 Bytes (0x00 00) / OUT 1 Byte (0x00)	SUBSCRIBE: "Input_Byte_IB200" (0x00) / Control (0x00) / Status (0x00)	
Slot#: 13	Input Word	IN 3 Bytes (0x00 00 00) / OUT 1 Byte (0x00)	SUBSCRIBE: "Controlword" (0x00 00) / Control (0x00) / Status (0x00)	

12.2 Export/Import of the configuration

The settings made on the web page can be saved on the PC in an editable format for backup, for series production or for manual editing of the device configuration (download).

If required, the file for configuring a device can then be uploaded again.

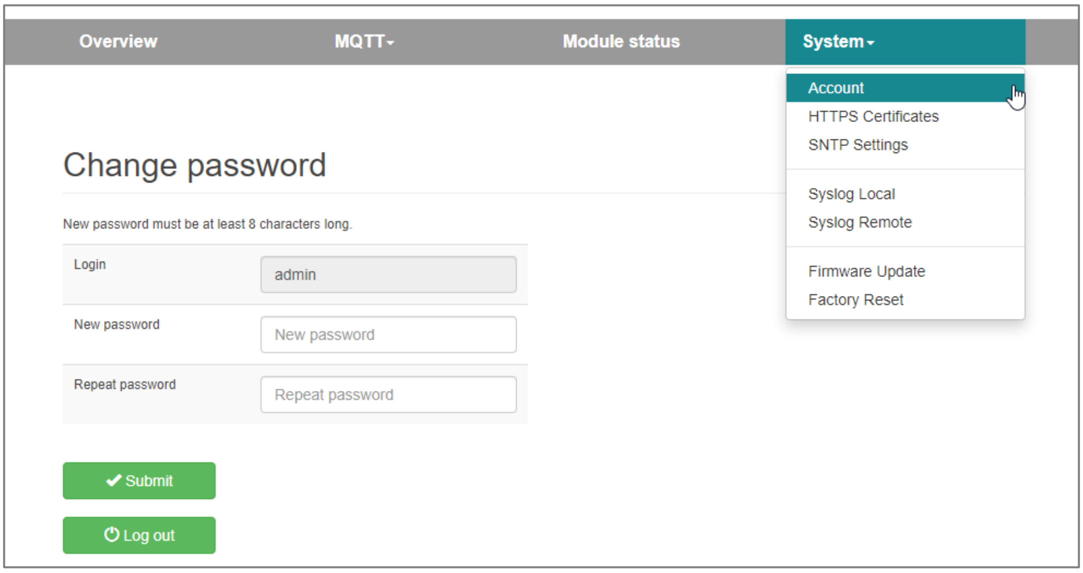


Configuration File

Upload or download configuration file of the MQTT

12.3 Account

In the menu "System" under "Account" the password of the user "admin" can be changed.



Change password

New password must be at least 8 characters long.

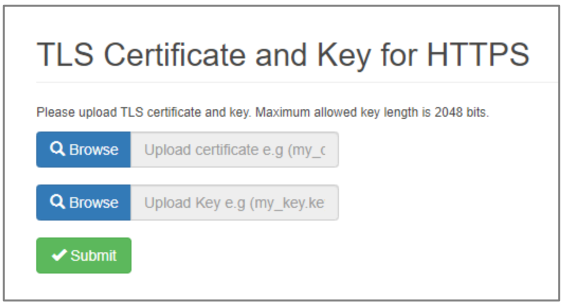
Login	admin
New password	New password
Repeat password	Repeat password

Currently the PN/MQTT Coupler contains only this user, the name is not changeable.

12.4 Upload TLS certificates for HTTPS

For secure access to the PN/MQTT Coupler website, a company certificate can be stored in the "System" menu under "HTTPS Certificates".

This ensures that the call to the PN/MQTT Coupler configuration web page is trusted in addition to HTTPS encryption.

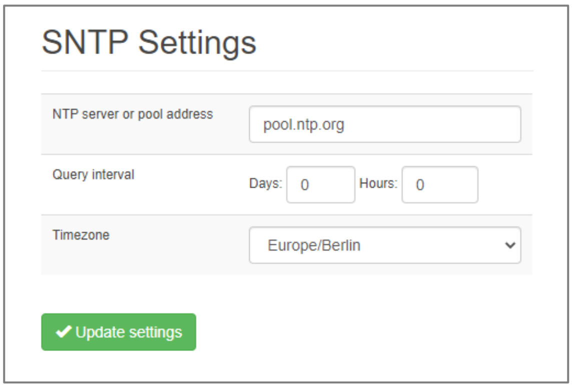


TLS Certificate and Key for HTTPS

Please upload TLS certificate and key. Maximum allowed key length is 2048 bits.

12.5 SNTP Settings

The PN/MQTT coupler can update its time via the MQTT network interface (X2) using the SNTP protocol. The time is used for checking the Certificates, logging or for timestamps in the MQTT topics.



SNTP Settings

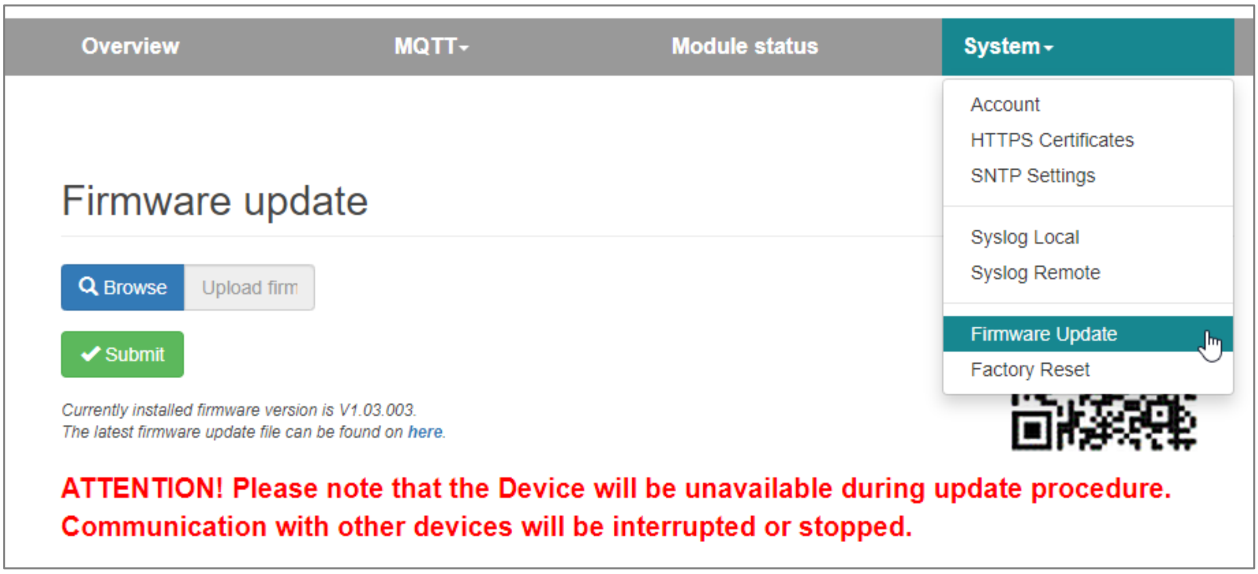
NTP server or pool address	pool.ntp.org
Query interval	Days: 0 Hours: 0
Timezone	Europe/Berlin

12.6 Firmware update

The firmware of the PN/MQTT Coupler can be updated very easily via the website. You can obtain the firmware from the Helmholz website at www.helmholz.de.

Link to latest firmware:
http://www.helmholz.de/goto/700-162-3MQ02#tab-software

The firmware file can be recognized by the file extension "HUF" (Helmholz Update File) and is encrypted to protect it from modification.



Firmware update

Currently installed firmware version is V1.03.003.
The latest firmware update file can be found on here.

ATTENTION! Please note that the Device will be unavailable during update procedure. Communication with other devices will be interrupted or stopped.

Place the firmware file on your PC and select the storage location with "Browse" and start the firmware update with "Submit". The firmware file is then transferred, decrypted and checked. If the content is correct, the firmware is burned into the program memory and the PN/MQTT Coupler is restarted.

ATTENTION During the update process the operation of the PN/MQTT Coupler is interrupted. Do not switch off the device during the update process!

NOTE The configuration of the PN/MQTT Coupler will be kept when updating to a higher version, as far as it is technically possible. However, a "downgrade" to an older firmware version can lead to configuration errors. It is recommended to perform a factory reset after a downgrade.

12.7 Reset to factory settings

Resetting the PN/MQTT Coupler to factory settings can be performed via the website or via the PROFINET function.

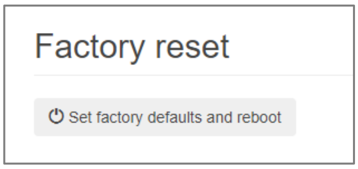
When resetting the PN/MQTT Coupler, the configuration is irretrievably deleted and the settings are set to the delivery state. The firmware remains at the current state.

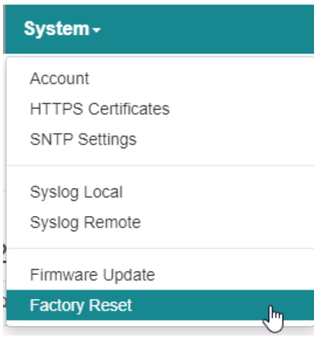
> ACHTUNG Please note that the device is no longer available in the network after resetting to factory settings. The PROFINET name and IP addresses are deleted, communication with the PLC is stopped and the PLC detects a configuration error and may also go to stop.

12.7.1 Reset to factory settings via website

Select the "Factory Reset" menu item in the "System" menu.

Press the button "Set factory defaults and reboot" and confirm the security prompt.

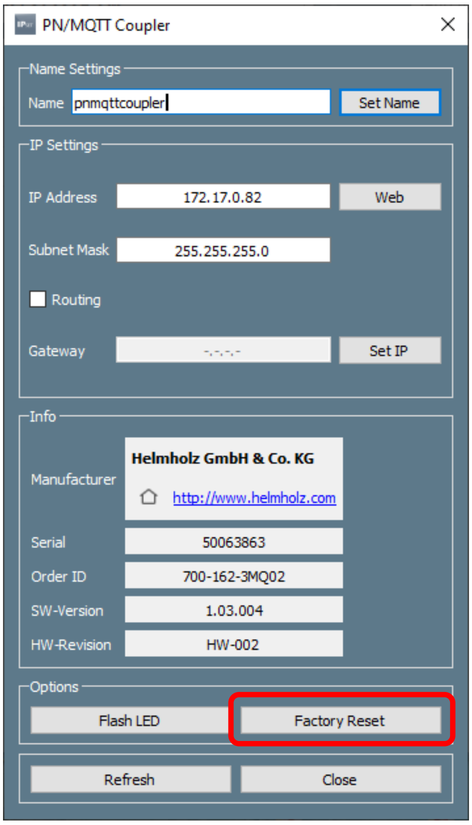

Factory reset



12.7.2 Reset to factory settings via "IPSet" tool

To reset the PN/MQTT Coupler to factory settings, the Helmholz tool "IPSet" can also be used via the PROFINET network.

The Helmholz IPSet Tool can be downloaded free of charge from the Helmholz website at the product or scan the QR code.


PN/MQTT Coupler

Name Settings

Name: pnmqttcoupler

IP Settings

IP Address: 172.17.0.82

Subnet Mask: 255.255.255.0

□ Routing

Gateway: -.-.-.-

Info

Manufacturer: Helmholz GmbH & Co. KG http://www.helmholz.com

Serial: 50063863

Order ID: 700-162-3MQ02

SW-Version: 1.03.004

HW-Revision: HW-002

Options

13 Diagnosis via LEDs

	X1 PROFINET (left side)	X2 MQTT network (right side)
SF (red)		
Off	Configuration correct	Configuration correct
On	PROFINET diagnostic alarm pending	PROFINET side not configured or failed
Flashing	PROFINET function "LED flashing" for finding the device is executed	-
BF (red)		
Off	Connection to PROFINET controller is established	MQTT broker connection is active
On	The device has no configuration, the PROFINET device name is incorrect, or there is no connection with the PROFINET controller	No connection to the MQTT broker can be established
Flashing	PROFINET function "LED flashing" for finding the device is executed	-
MT (yellow)		
Flashing	A firmware update is being carried out	A firmware update is being carried out
Flashing with SF and BF	PROFINET function “LED flashing” for finding the device is being carried out	-
PWR (green)		
On	PS1 Power supply present	PS2 Power supply present
RUN (green)		
Off	Firmware or device defective. Please contact Support	
On	The device is ready to operate	
RJ45 LEDs	X1 P1/P2 und X2 P1/P2	
Green (Link)	Connected	
Orange (Act)	Data transfer at the port active	

14 Client tools for MQTT data exchange

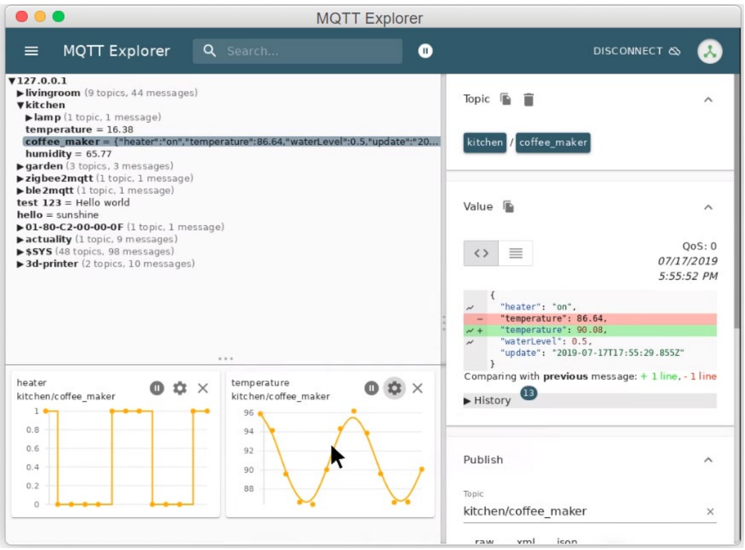
In order to be able to test the data exchange via the PN/MQTT coupler when the application is not yet available on the other side or to be able to observe the data sent by the PN/MQTT coupler next to the application, the use of MQTT client test programs is recommended. In this chapter 3 MQTT clients are presented as examples, but there are many more tools.

14.1 MQTT Explorer

The "MQTT Explorer" by Thomas Nordquist (http://mqtt-explorer.com/) is a very practical little tool for Windows, Mac and Linux.

Besides a tidy interface, hierarchical display of topics and the possibility to display values in graphs, the program works very fast and is compact.

GitHub:



```
▼127.0.0.1
  ▶livingroom (9 topics, 44 messages)
  ▼kitchen
    ▶lamp (1 topic, 1 message)
    temperature = 16.38
    coffee_maker = {"heater":"on","temperature":86.64,"waterLevel":0.5,"update":"20...
    humidity = 65.77
  ▶garden (3 topics, 3 messages)
  ▶zigbee2mqtt (1 topic, 1 message)
  ▶ble2mqtt (1 topic, 1 message)
  test 123 = Hello world
  hello = sunshine
  ▶01-80-C2-00-00-0F (1 topic, 1 message)
  ▶actuality (1 topic, 9 messages)
  ▶$SYS (48 topics, 98 messages)
  ▶3d-printer (2 topics, 10 messages)
```

QoS: 0
07/17/2019
5:55:52 PM

```
{
    "heater": "on",
-   "temperature": 86.64,
+   "temperature": 90.08,
    "waterLevel": 0.5,
    "update": "2019-07-17T17:55:29.855Z"
}
```

Comparing with **previous** message: + 1 line, - 1 line

heater kitchen/coffee_maker

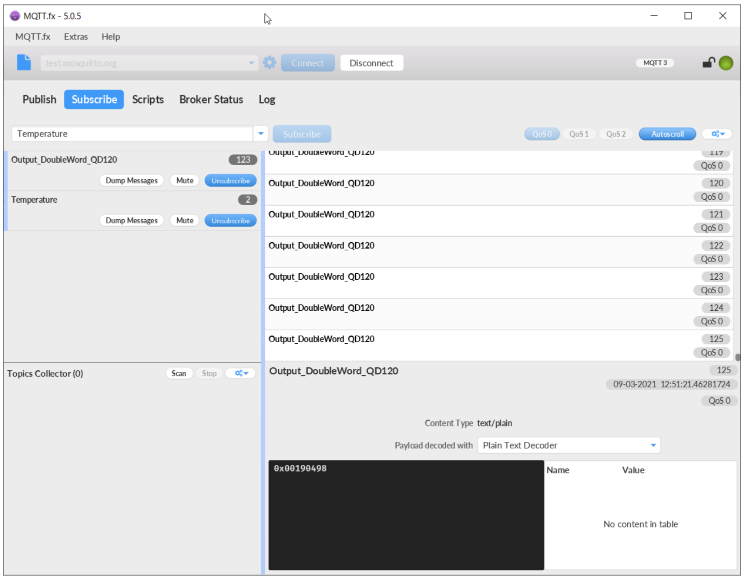
temperature kitchen/coffee_maker

Topic: kitchen/coffee_maker

14.2 MQTT.fx V5

MQTT.fx offers a very extensive range of functions in the paid version from version V5.

In particular, the extensible payload decoders facilitate the testing with MQTT.



test.mosquitto.org

Temperature

Output_DoubleWord_QD120 123

Temperature 2

Topics Collector (0)

Output_DoubleWord_QD120	119 QoS 0
Output_DoubleWord_QD120	120 QoS 0
Output_DoubleWord_QD120	121 QoS 0
Output_DoubleWord_QD120	122 QoS 0
Output_DoubleWord_QD120	123 QoS 0
Output_DoubleWord_QD120	124 QoS 0
Output_DoubleWord_QD120	125 QoS 0

Output_DoubleWord_QD120 125
09-03-2021 12:51:21.46281724
QoS 0

Content Type text/plain

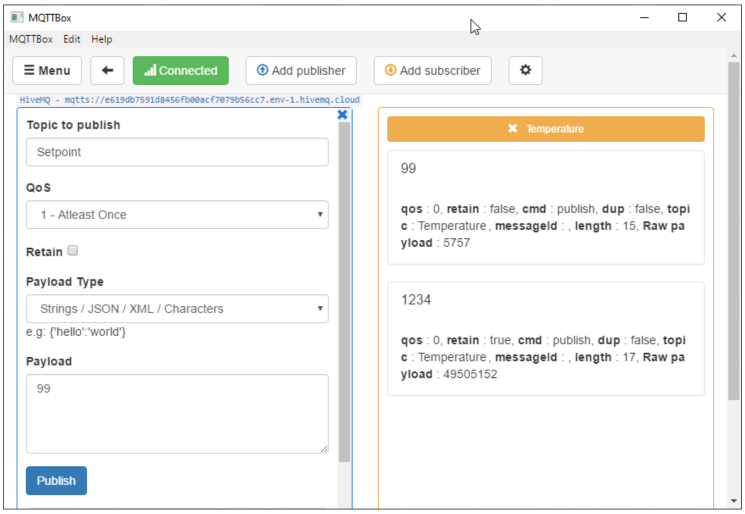
Payload decoded with Plain Text Decoder

```
0x00190498
```

Name	Value
No content in table	

14.3 MQTT Box

MQTT Box (http://workswithweb.com/mqttbox.html) is easily installable as a Linux, Macintosh or Windows APP.



HiveMQ - mqtts://e619db7591d8456fb00acf7079b56cc7.env-1.hivemq.cloud

Topic to publish: Setpoint

QoS: 1 - Atleast Once

Retain ☐

Payload Type: Strings / JSON / XML / Characters

e.g: {'hello':'world'}

Payload: 99

Temperature

99

qos : 0, **retain** : false, **cmd** : publish, **dup** : false, **topic** : Temperature , **messageId** : , **length** : 15, **Raw payload** : 5757

1234

qos : 0, **retain** : true, **cmd** : publish, **dup** : false, **topic** : Temperature , **messageId** : , **length** : 17, **Raw payload** : 49505152

15 Application example „mosquitto“

Eclipse Mosquitto (mosquitto.org) is an open source MQTT broker for MQTT V3.1.1 and MQTT V5.

Mosquitto is available for Linux - including the Raspberry PI - as well as for the PC. The Mosquitto project also includes a command line MQTT client for testing.

15.1 Mosquitto Test-Host

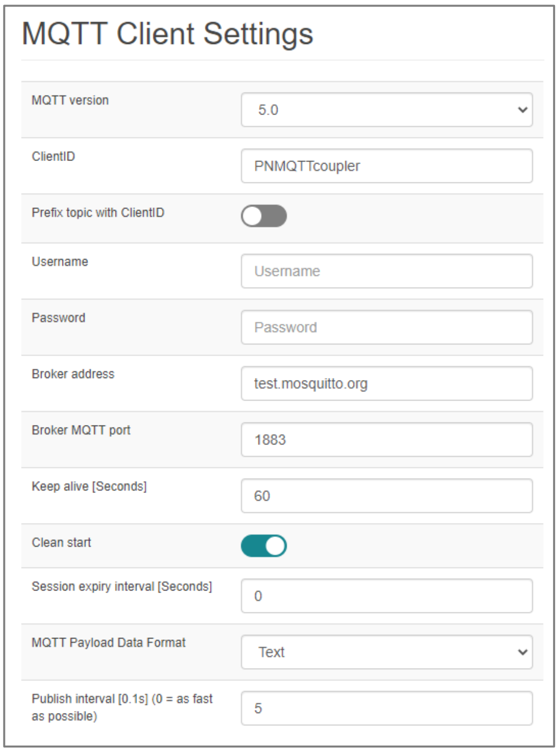
Eclipse Mosquitto operates a Mosquitto test broker at test.mosquitto.org.

The test broker can be addressed with MQTT V3.1.1 as well as with MQTT V5.

The easiest way to reach the test broker is unencrypted (port 1883).

Details on usage are explained on the Mosquitto test broker website ↗.

Note: To be able to resolve the domain name of the test broker, a valid DNS server must be specified in the IP settings!



MQTT Client Settings

MQTT version	5.0
ClientID	PNMQTTcoupler
Prefix topic with ClientID	○
Username	Username
Password	Password
Broker address	test.mosquitto.org
Broker MQTT port	1883
Keep alive [Seconds]	60
Clean start	●
Session expiry interval [Seconds]	0
MQTT Payload Data Format	Text
Publish interval [0.1s] (0 = as fast as possible)	5

15.2 Install and use Mosquitto locally

Mosquitto can run on Linux, on a Windows PC (64-bit and 32-bit) and on a Macintosh. A wide variety of derivatives are supported under Linux: Raspberry PI, Debian, Ubuntu, etc.

The corresponding packages can be downloaded here ↗ or installed with a package manager.

16 Application example „HiveMQ“

HiveMQ (https://www.hivemq.com) is a professional, scalable MQTT broker that runs both locally on the PC (Windows or Linux) and can be used in the cloud with high performance and high availability. HiveMQ can be used as a broker between the device layer and the cloud applications (AWS, Azure, SAP, etc.).

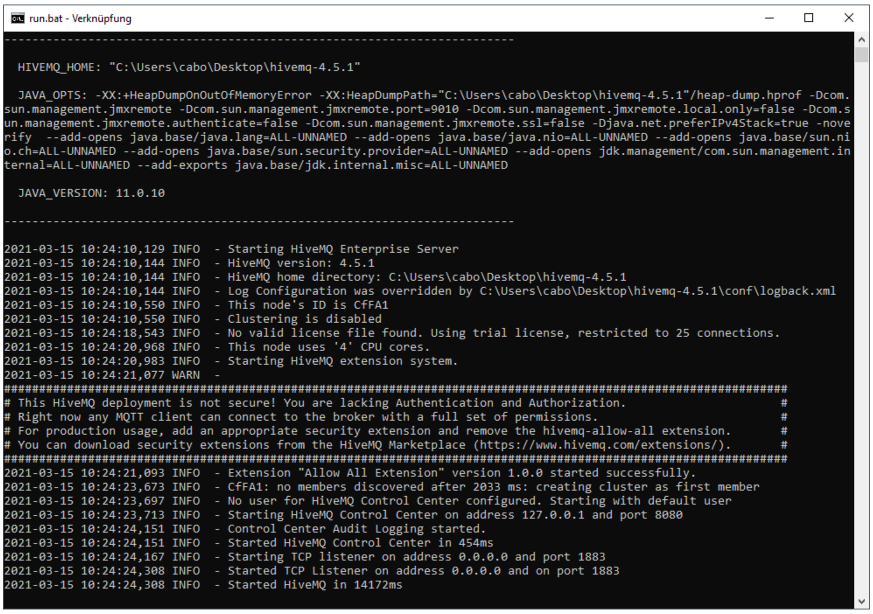
HiveMQ is available in 3 variants: "Community" (open source on GitHub), "Professional" and "Enterprise". The last two versions can be run on your own servers (on premise) or used as a cloud service by HiveMQ.

16.1 Use of HiveMQ in a virtual machine

An "out-of-the-box" trial version of the enterprise version HiveMQ solution can be downloaded and launched at https://www.hivemq.com/downloads/. The version can be launched on Windows, Linux. A Docker version or a variant that can be launched directly in AWS is also available.

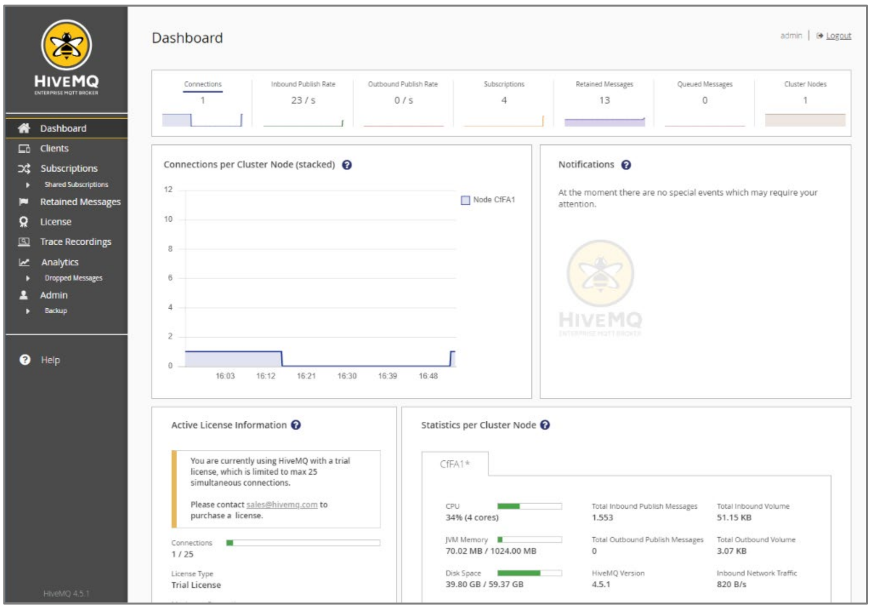
The trial version can be accessed directly from the PN/MQTT coupler under the IP address of the PC under which it was started.

The MQTT port 1833 is active, no encryption or "user/password" is used.



```
-------------------------------------------------------------------------

  HIVEMQ_HOME: "C:\Users\cabo\Desktop\hivemq-4.5.1"

  JAVA_OPTS: -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath="C:\Users\cabo\Desktop\hivemq-4.5.1"/heap-dump.hprof -Dcom.
sun.management.jmxremote -Dcom.sun.management.jmxremote.port=9010 -Dcom.sun.management.jmxremote.local.only=false -Dcom.s
un.management.jmxremote.authenticate=false -Dcom.sun.management.jmxremote.ssl=false -Djava.net.preferIPv4Stack=true -nove
rify  --add-opens java.base/java.lang=ALL-UNNAMED --add-opens java.base/java.nio=ALL-UNNAMED --add-opens java.base/sun.ni
o.ch=ALL-UNNAMED --add-opens java.base/sun.security.provider=ALL-UNNAMED --add-opens jdk.management/com.sun.management.in
ternal=ALL-UNNAMED --add-exports java.base/jdk.internal.misc=ALL-UNNAMED

  JAVA_VERSION: 11.0.10

-------------------------------------------------------------------------

2021-03-15 10:24:10,129 INFO  - Starting HiveMQ Enterprise Server
2021-03-15 10:24:10,144 INFO  - HiveMQ version: 4.5.1
2021-03-15 10:24:10,144 INFO  - HiveMQ home directory: C:\Users\cabo\Desktop\hivemq-4.5.1
2021-03-15 10:24:10,144 INFO  - Log Configuration was overridden by C:\Users\cabo\Desktop\hivemq-4.5.1\conf\logback.xml
2021-03-15 10:24:10,550 INFO  - This node's ID is CfFA1
2021-03-15 10:24:10,550 INFO  - Clustering is disabled
2021-03-15 10:24:18,543 INFO  - No valid license file found. Using trial license, restricted to 25 connections.
2021-03-15 10:24:20,968 INFO  - This node uses '4' CPU cores.
2021-03-15 10:24:20,983 INFO  - Starting HiveMQ extension system.
2021-03-15 10:24:21,077 WARN  -
################################################################################################
# This HiveMQ deployment is not secure! You are lacking Authentication and Authorization.      #
# Right now any MQTT client can connect to the broker with a full set of permissions.          #
# For production usage, add an appropriate security extension and remove the hivemq-allow-all extension.      #
# You can download security extensions from the HiveMQ Marketplace (https://www.hivemq.com/extensions/).      #
################################################################################################
2021-03-15 10:24:21,093 INFO  - Extension "Allow All Extension" version 1.0.0 started successfully.
2021-03-15 10:24:23,673 INFO  - CfFA1: no members discovered after 2033 ms: creating cluster as first member
2021-03-15 10:24:23,697 INFO  - No user for HiveMQ Control Center configured. Starting with default user
2021-03-15 10:24:23,713 INFO  - Starting HiveMQ Control Center on address 127.0.0.1 and port 8080
2021-03-15 10:24:24,151 INFO  - Control Center Audit Logging started.
2021-03-15 10:24:24,151 INFO  - Started HiveMQ Control Center in 454ms
2021-03-15 10:24:24,167 INFO  - Starting TCP listener on address 0.0.0.0 and port 1883
2021-03-15 10:24:24,308 INFO  - Started TCP Listener on address 0.0.0.0 and on port 1883
2021-03-15 10:24:24,308 INFO  - Started HiveMQ in 14172ms
```

The trial version also includes an extensive information web page.



Dashboard

Connections	Inbound Publish Rate	Outbound Publish Rate	Subscriptions	Retained Messages	Queued Messages	Cluster Nodes
1	23 / s	0 / s	4	13	0	1

Connections per Cluster Node (stacked)

Notifications

At the moment there are no special events which may require your attention.

Active License Information

You are currently using HiveMQ with a trial license, which is limited to max 25 simultaneous connections.

Please contact sales@hivemq.com to purchase a license.

Connections
1 / 25

License Type
Trial License

Statistics per Cluster Node

CPU
34% (4 cores)

Total Inbound Publish Messages
1.553

Total Inbound Volume
51.15 KB

JVM Memory
70.02 MB / 1024.00 MB

Total Outbound Publish Messages
0

Total Outbound Volume
3.07 KB

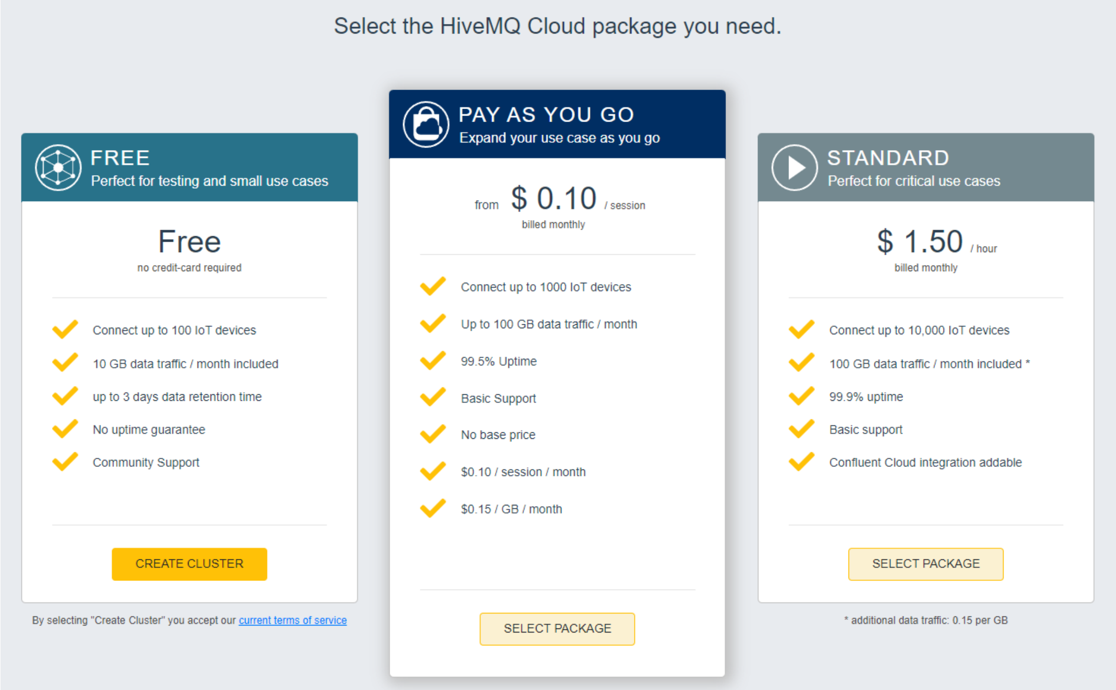
Disk Space
39.80 GB / 59.37 GB

HiveMQ Version
4.5.1

Inbound Network Traffic
820 B/s

16.2 HiveMQ Cloud

At https://www.hivemq.com/cloud/, an account can be created for the HiveMQ Cloud and own "clusters" (MQTT brokers) can be operated.



Select the HiveMQ Cloud package you need.

FREE
Perfect for testing and small use cases

Free
no credit-card required

- ✓ Connect up to 100 IoT devices
- ✓ 10 GB data traffic / month included
- ✓ up to 3 days data retention time
- ✓ No uptime guarantee
- ✓ Community Support

By selecting "Create Cluster" you accept our current terms of service

PAY AS YOU GO
Expand your use case as you go

from $ 0.10 / session
billed monthly

- ✓ Connect up to 1000 IoT devices
- ✓ Up to 100 GB data traffic / month
- ✓ 99.5% Uptime
- ✓ Basic Support
- ✓ No base price
- ✓ $0.10 / session / month
- ✓ $0.15 / GB / month

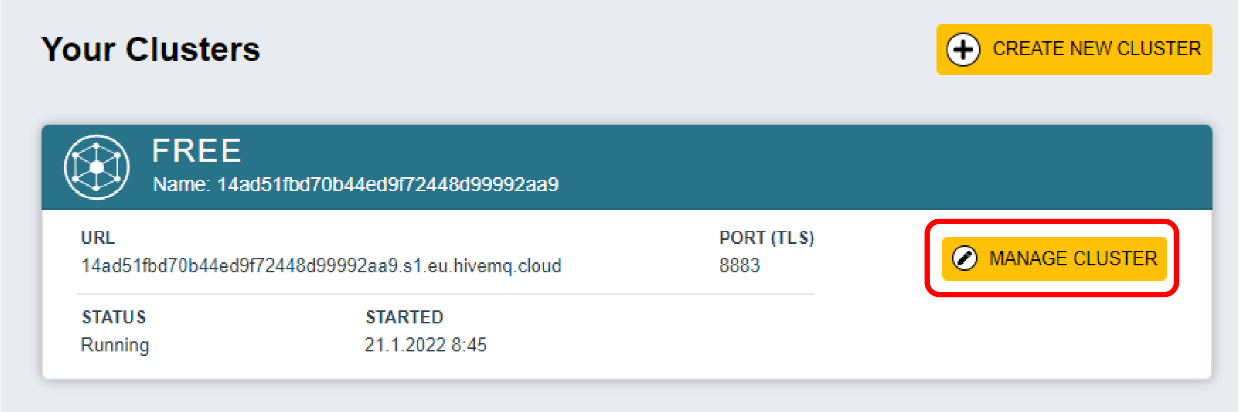
STANDARD
Perfect for critical use cases

$ 1.50 / hour
billed monthly

- ✓ Connect up to 10,000 IoT devices
- ✓ 100 GB data traffic / month included *
- ✓ 99.9% uptime
- ✓ Basic support
- ✓ Confluent Cloud integration addable

* additional data traffic: 0.15 per GB

HiveMQ offers the setup of a free cluster in the "Free" model for testing and small use cases. In addition, you can choose from two paid models for professional operation.



Your Clusters

FREE
Name: 14ad51fbd70b44ed9f72448d99992aa9

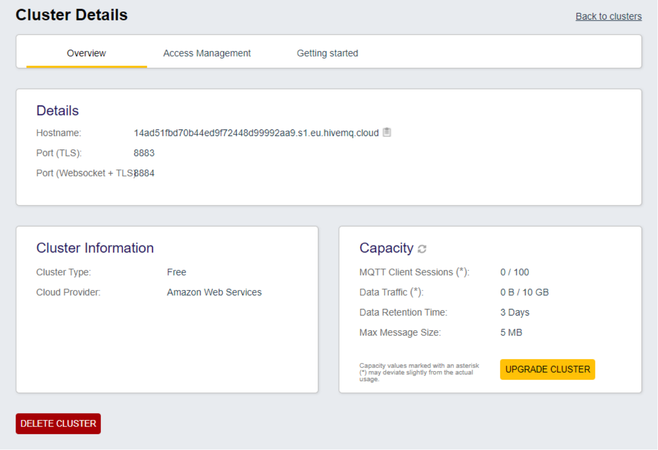
URL	PORT (TLS)
14ad51fbd70b44ed9f72448d99992aa9.s1.eu.hivemq.cloud	8883

STATUS	STARTED
Running	21.1.2022 8:45

You can access the "Cluster Detail" via "Manage Cluster".

In this dialog, click the "Access Management" tab to create a new MQTT client access ("Username"/"Password") under "MQTT Credentials" with "Add".

Herewith all necessary settings are already done and the PN/MQTT Coupler can access the HiveMQ Broker.



Cluster Details

Details

Hostname:	14ad51fbd70b44ed9f72448d99992aa9.s1.eu.hivemq.cloud
Port (TLS):	8883
Port (Websocket + TLS):	8884

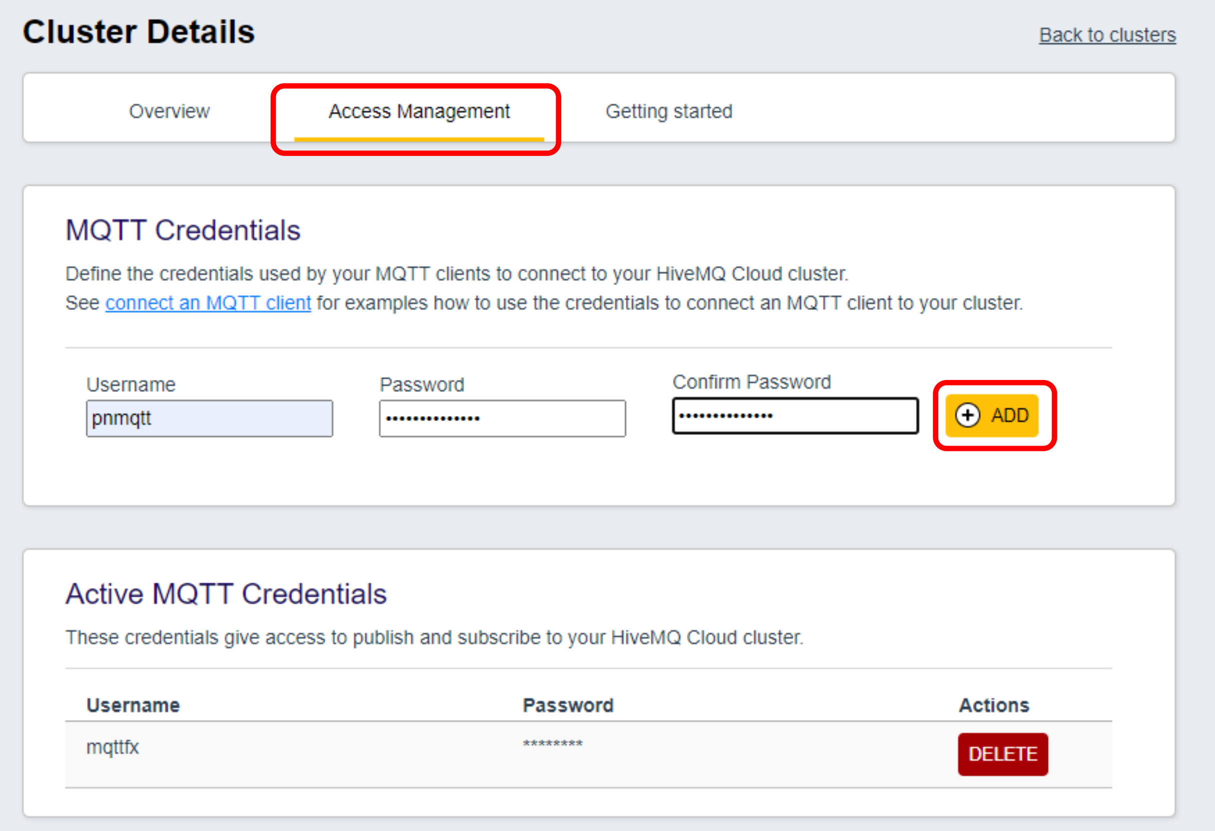
Cluster Information

Cluster Type:	Free
Cloud Provider:	Amazon Web Services

Capacity

MQTT Client Sessions (*):	0 / 100
Data Traffic (*):	0 B / 10 GB
Data Retention Time:	3 Days
Max Message Size:	5 MB

Capacity values marked with an asterisk (*) may deviate slightly from the actual usage.



Cluster Details

MQTT Credentials

Define the credentials used by your MQTT clients to connect to your HiveMQ Cloud cluster.
See connect an MQTT client for examples how to use the credentials to connect an MQTT client to your cluster.

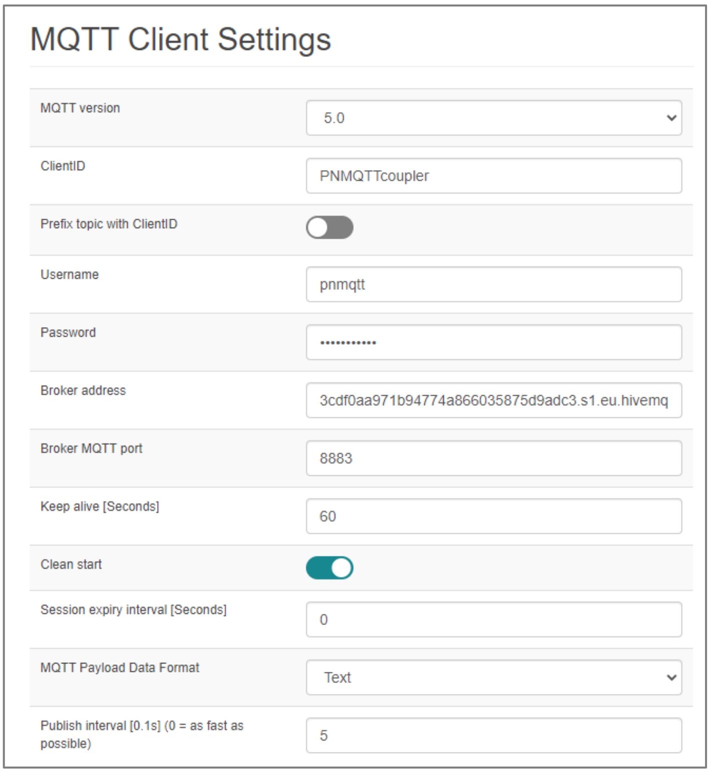
Username	Password	Confirm Password	
pnmqtt	•••••••••••••••	•••••••••••••••	ADD

Active MQTT Credentials

These credentials give access to publish and subscribe to your HiveMQ Cloud cluster.

Username	Password	Actions
mqttfx	********	DELETE

In the PN/MQTT Coupler, the following settings must now be applied under "MQTT Client Settings":



MQTT Client Settings

MQTT version	5.0
ClientID	PNMQTTcoupler
Prefix topic with ClientID	○
Username	pnmqtt
Password	•••••••••••
Broker address	3cdf0aa971b94774a866035875d9adc3.s1.eu.hivemq
Broker MQTT port	8883
Keep alive [Seconds]	60
Clean start	●
Session expiry interval [Seconds]	0
MQTT Payload Data Format	Text
Publish interval [0.1s] (0 = as fast as possible)	5

HiveMQ supports both **MQTT version** 3.1.1 and version 5.

The **ClientID** can be any.

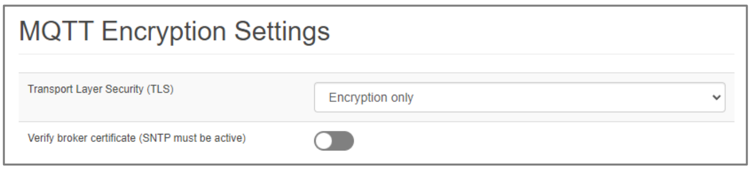
Under **Username** and **Password**, accept the values entered in the MQTT Credentials.

Under **Broker address** copy the HiveMQ URL.

Set **Broker Port** to 8883.

The remaining settings can be selected as desired.

Transport Layer Security (TLS) must be set to "Encryption only".

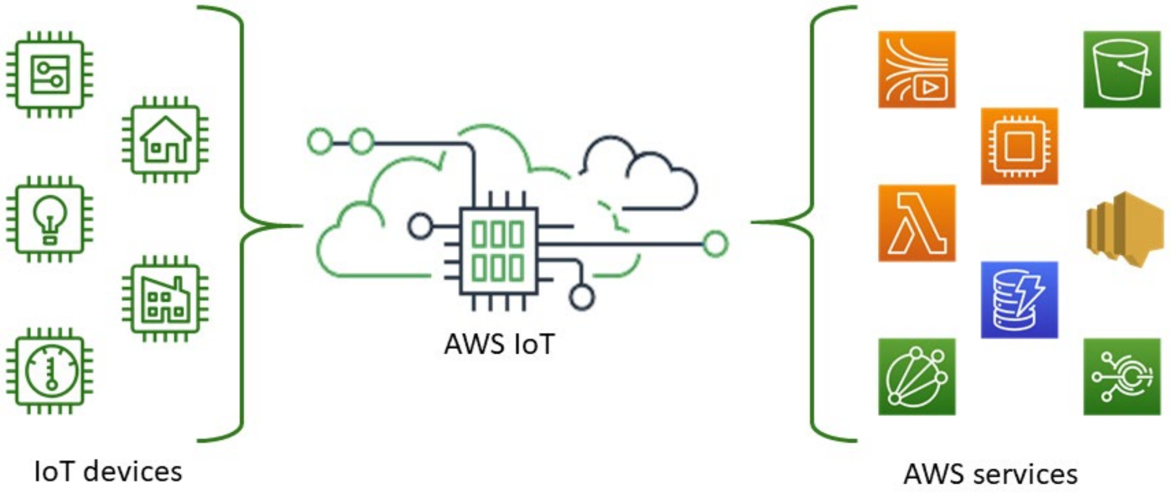


MQTT Encryption Settings

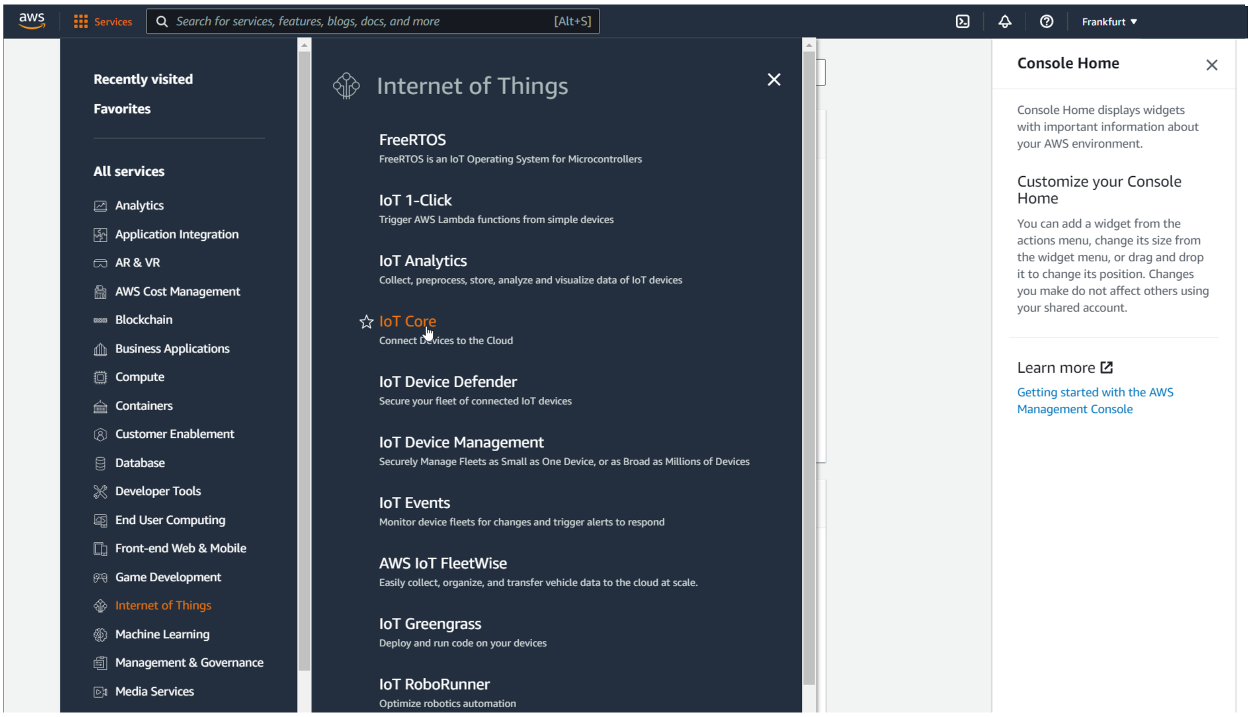
Transport Layer Security (TLS)	Encryption only
Verify broker certificate (SNTP must be active)	○

17 Application example „Amazon IoT Core“

The PN/MQTT Coupler makes it very easy to transfer data directly to the Amazon Cloud (AWS). The AWS IoT Core component is an MQTT broker in the AWS Cloud. MQTT messages can be sent directly to AWS IoT Core and then processed in the other AWS services.



In your AWS account, select Internet of Things / IoT Core module.



FreeRTOS
FreeRTOS is an IoT Operating System for Microcontrollers

IoT 1-Click
Trigger AWS Lambda functions from simple devices

IoT Analytics
Collect, preprocess, store, analyze and visualize data of IoT devices

☆ IoT Core
Connect Devices to the Cloud

IoT Device Defender
Secure your fleet of connected IoT devices

IoT Device Management
Securely Manage Fleets as Small as One Device, or as Broad as Millions of Devices

IoT Events
Monitor device fleets for changes and trigger alerts to respond

AWS IoT FleetWise
Easily collect, organize, and transfer vehicle data to the cloud at scale.

IoT Greengrass
Deploy and run code on your devices

IoT RoboRunner
Optimize robotics automation

Console Home ✕

Console Home displays widgets with important information about your AWS environment.

Customize your Console Home

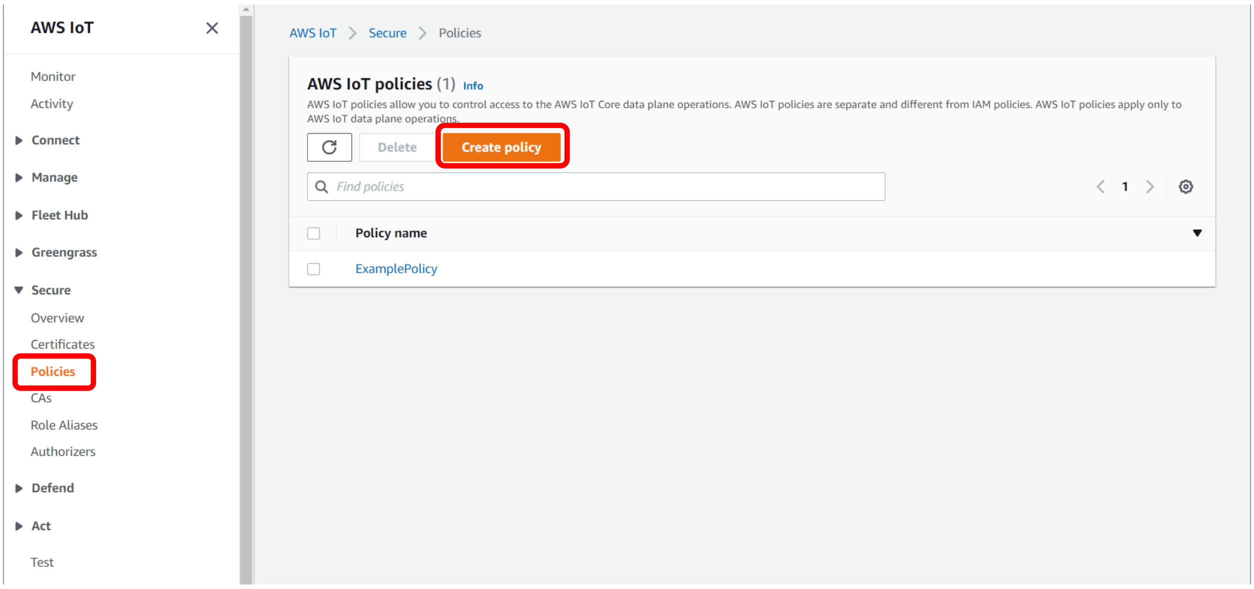
You can add a widget from the actions menu, change its size from the widget menu, or drag and drop it to change its position. Changes you make do not affect others using your shared account.

Learn more

Getting started with the AWS Management Console

17.1 Create a policy

Create a new device policy (set of rules for access rights) for the PN/MQTT Coupler under "Secure/Policies" and assign a name for the policy.



AWS IoT

Monitor

Activity

▶ Connect

▶ Manage

▶ Fleet Hub

▶ Greengrass

▼ Secure

Overview

Certificates

Policies

CAs

Role Aliases

Authorizers

▶ Defend

▶ Act

Test

AWS IoT > Secure > Policies

AWS IoT policies (1) Info

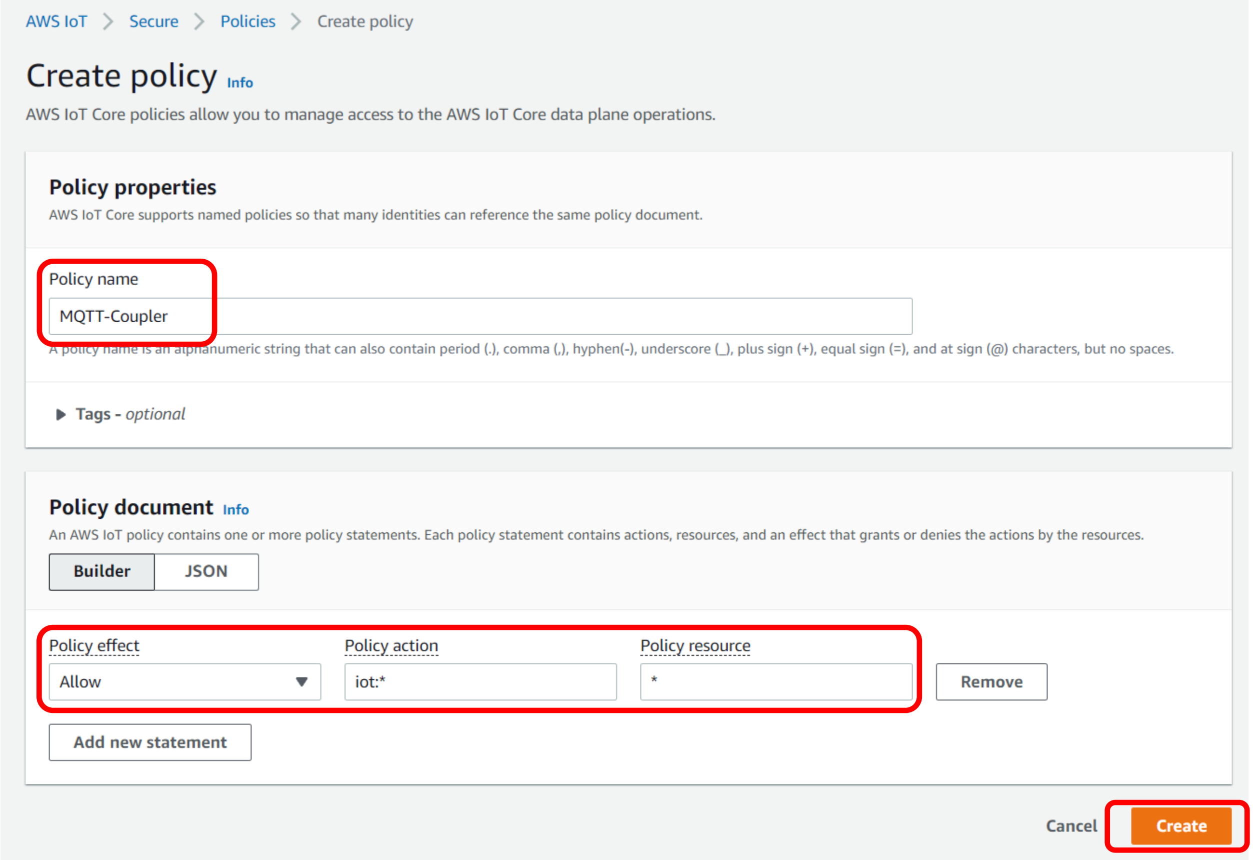
AWS IoT policies allow you to control access to the AWS IoT Core data plane operations. AWS IoT policies are separate and different from IAM policies. AWS IoT policies apply only to AWS IoT data plane operations.

Delete | Create policy

Find policies

□	Policy name
□	ExamplePolicy

Give the policy a name and enter "iot:*" for Action, "*" for Resource ARN and check "Allow" for Effect. Switch to the "Advanced mode" and check the display:



AWS IoT > Secure > Policies > Create policy

Create policy Info

AWS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.

Policy properties

AWS IoT Core supports named policies so that many identities can reference the same policy document.

Policy name: MQTT-Coupler

A policy name is an alphanumeric string that can also contain period (.), comma (,), hyphen(-), underscore (_), plus sign (+), equal sign (=), and at sign (@) characters, but no spaces.

▶ Tags - *optional*

Policy document Info

An AWS IoT policy contains one or more policy statements. Each policy statement contains actions, resources, and an effect that grants or denies the actions by the resources.

Builder | JSON

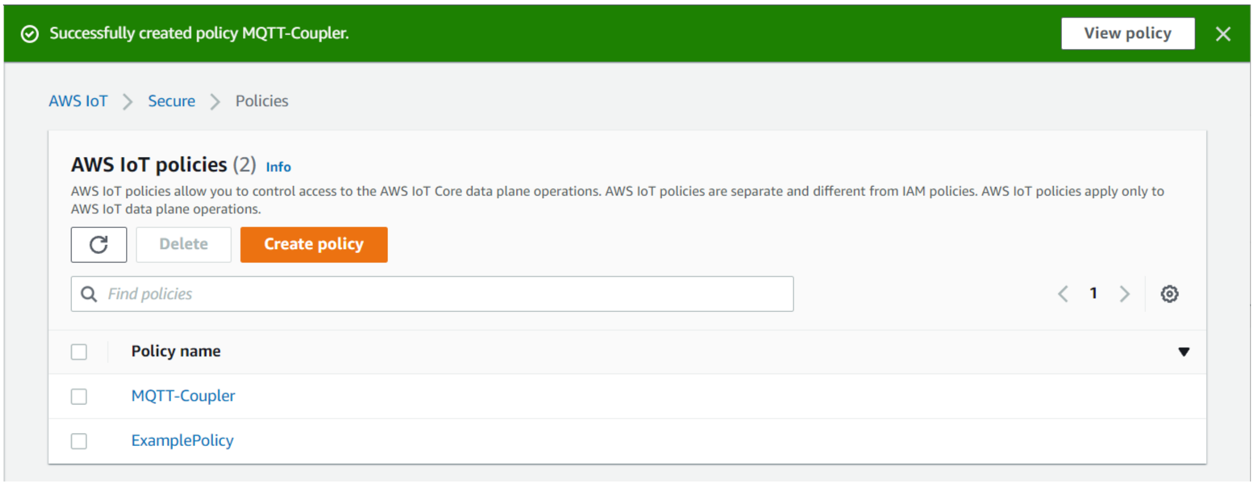
Policy effect	Policy action	Policy resource	
Allow	iot:*	*	Remove

Add new statement

Cancel | Create

This will give you full access to "Subscriptions" and "Publications" (can be customized later).

The policy is created with "Create" (bottom right) and appears in the overview.



Successfully created policy MQTT-Coupler.

AWS IoT > Secure > Policies

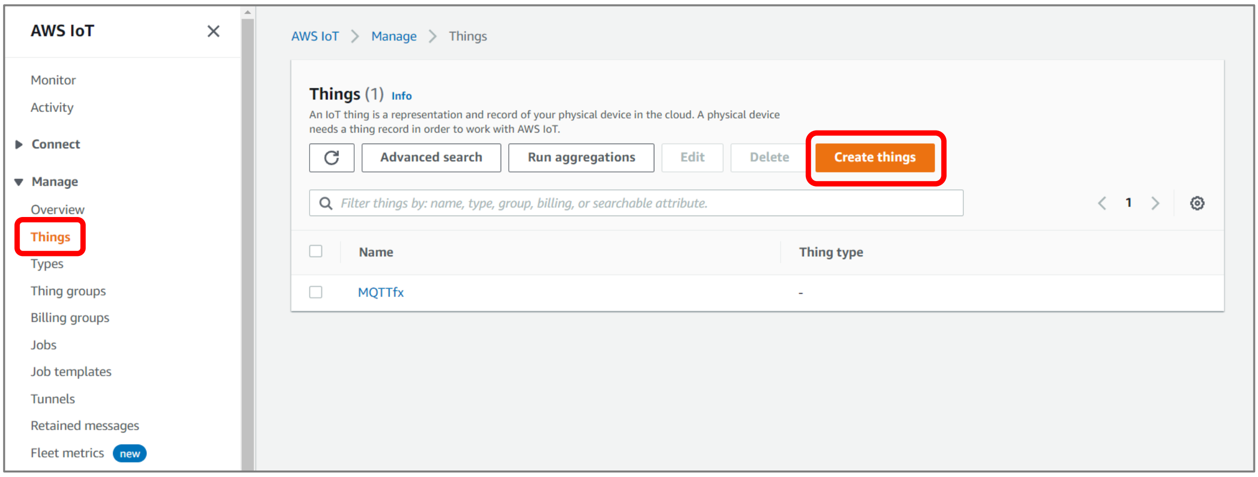
AWS IoT policies (2) Info

AWS IoT policies allow you to control access to the AWS IoT Core data plane operations. AWS IoT policies are separate and different from IAM policies. AWS IoT policies apply only to AWS IoT data plane operations.

Policy name
MQTT-Coupler
ExamplePolicy

17.2 Create „AWS IoT Things“

Under “Manage/Things”, select “Create a single thing”.



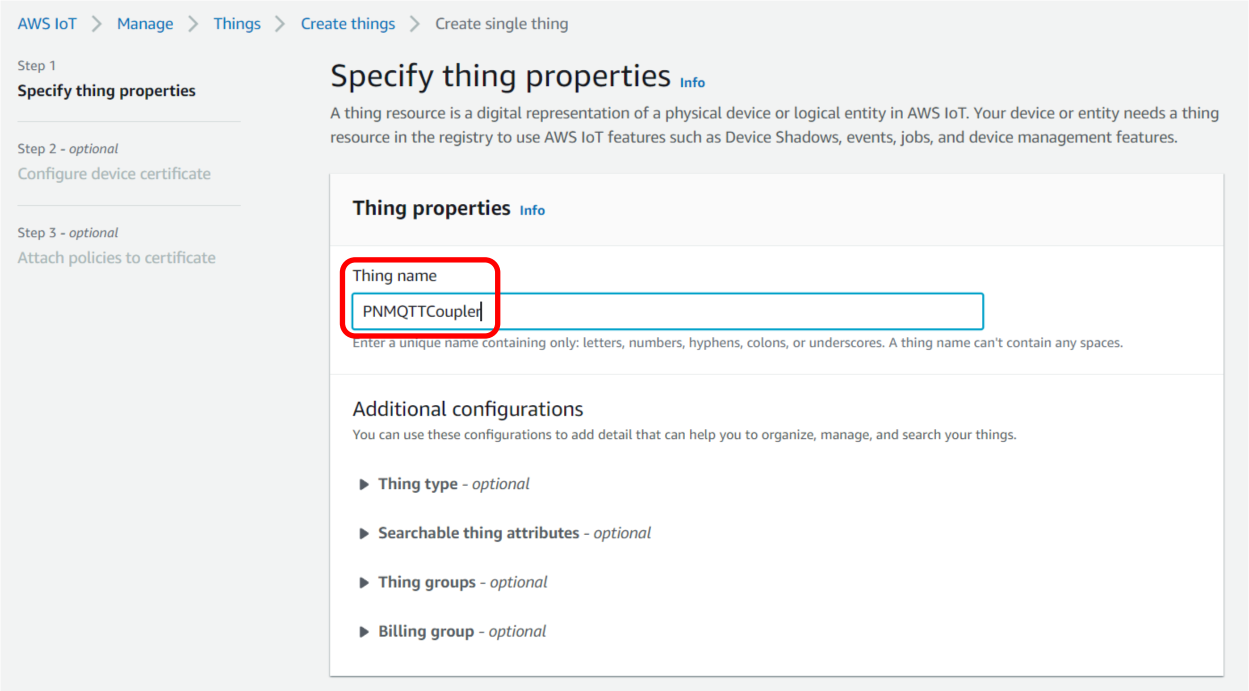
Things (1) Info

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

Filter things by: name, type, group, billing, or searchable attribute.

Name	Thing type
MQTTfx	-

Give the “Thing” a name.



AWS IoT > Manage > Things > Create things > Create single thing

Step 1
Specify thing properties

Step 2 - *optional*
Configure device certificate

Step 3 - *optional*
Attach policies to certificate

Specify thing properties Info

A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

Thing properties Info

Thing name: PNMQTTCoupler

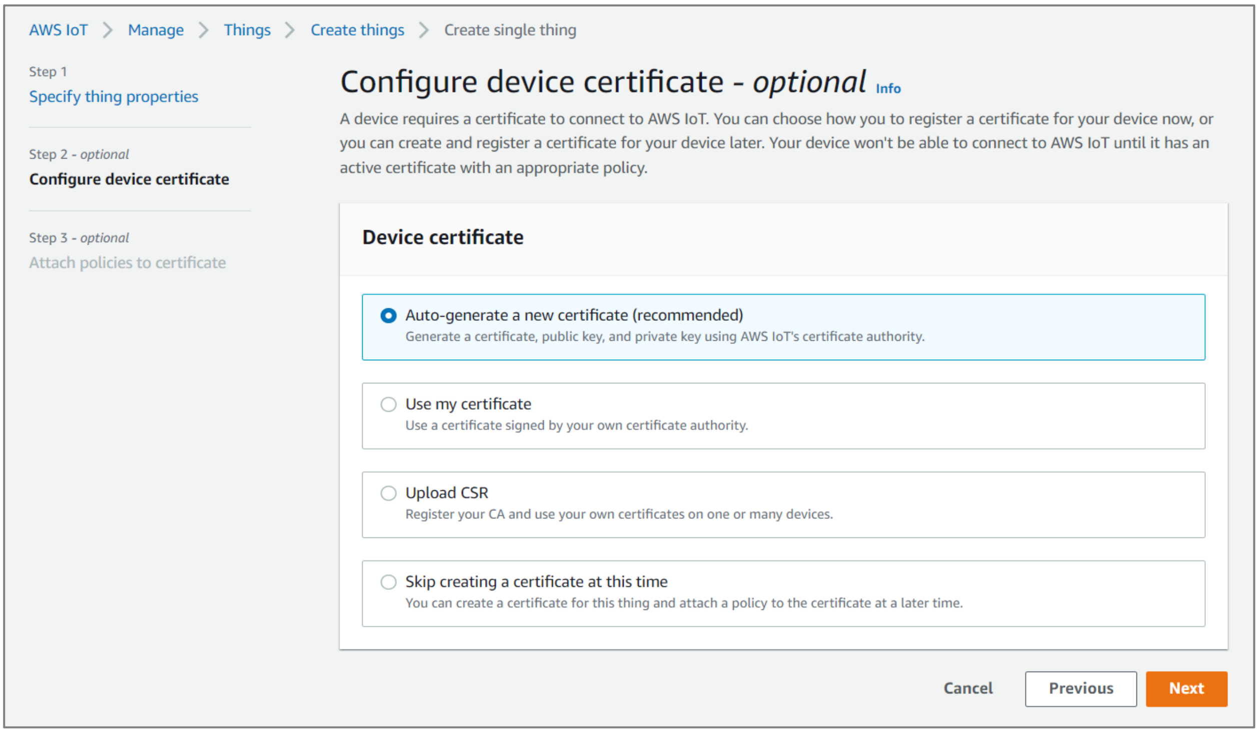
Enter a unique name containing only: letters, numbers, hyphens, colons, or underscores. A thing name can't contain any spaces.

Additional configurations

You can use these configurations to add detail that can help you to organize, manage, and search your things.

- **Thing type** - *optional*
- **Searchable thing attributes** - *optional*
- **Thing groups** - *optional*
- **Billing group** - *optional*

In the next dialog, select "Auto-generate a new certificate".



AWS IoT > Manage > Things > Create things > Create single thing

Step 1
Specify thing properties

Step 2 - *optional*
Configure device certificate

Step 3 - *optional*
Attach policies to certificate

Configure device certificate - *optional* Info

A device requires a certificate to connect to AWS IoT. You can choose how you to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.

Device certificate

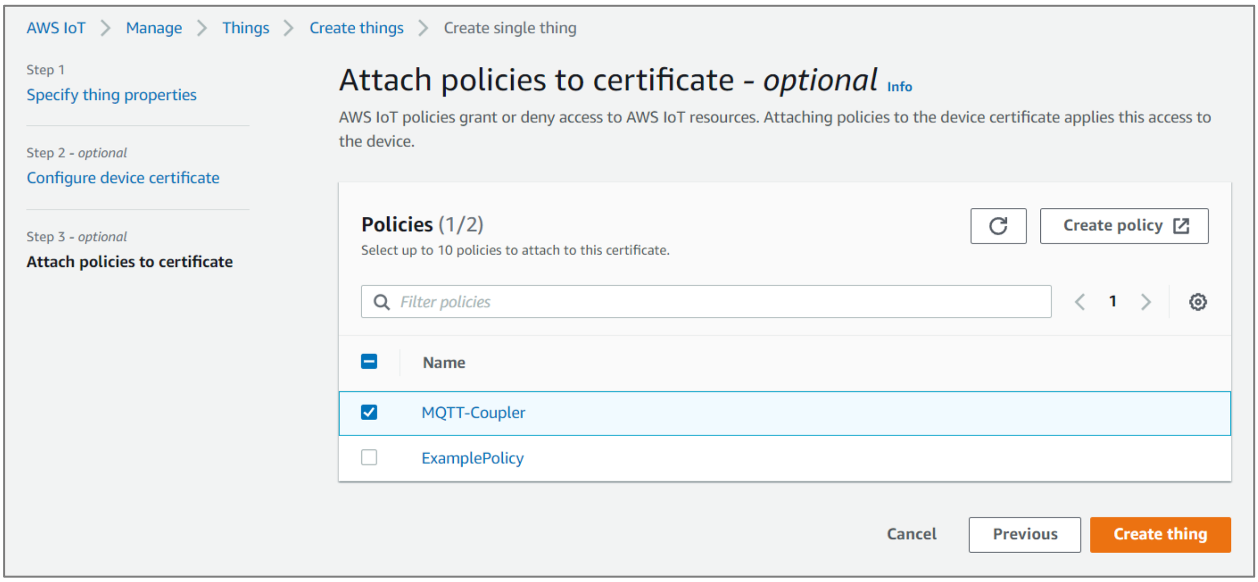
● Auto-generate a new certificate (recommended)
Generate a certificate, public key, and private key using AWS IoT's certificate authority.

○ Use my certificate
Use a certificate signed by your own certificate authority.

○ Upload CSR
Register your CA and use your own certificates on one or many devices.

○ Skip creating a certificate at this time
You can create a certificate for this thing and attach a policy to the certificate at a later time.

The certificate created for the object is now associated with the policy created above to give the object the necessary access rights.



AWS IoT > Manage > Things > Create things > Create single thing

Step 1
Specify thing properties

Step 2 - *optional*
Configure device certificate

Step 3 - *optional*
Attach policies to certificate

Attach policies to certificate - *optional* Info

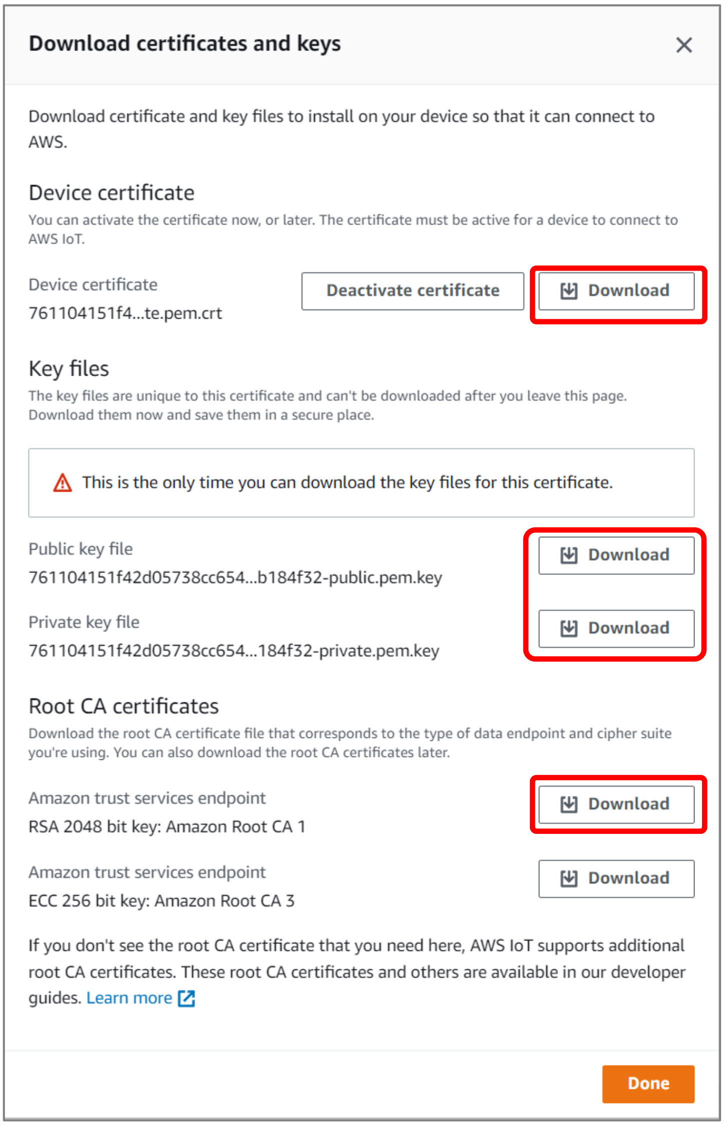
AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.

Policies (1/2)
Select up to 10 policies to attach to this certificate.

	Name
☑	MQTT-Coupler
□	ExamplePolicy

Download "Device certificate", "Public key file" and "Private key file" and keep them in a safe place, they cannot be reloaded.

Additionally, download the root certificate ("root CA") from AWS. For this sample application, use the "RSA 2048 bit key: Amazon Root CA 1".



Download certificates and keys

Download certificate and key files to install on your device so that it can connect to AWS.

Device certificate

You can activate the certificate now, or later. The certificate must be active for a device to connect to AWS IoT.

Device certificate
761104151f4...te.pem.crt

Key files

The key files are unique to this certificate and can't be downloaded after you leave this page. Download them now and save them in a secure place.

⚠ This is the only time you can download the key files for this certificate.

Public key file
761104151f42d05738cc654...b184f32-public.pem.key

Private key file
761104151f42d05738cc654...184f32-private.pem.key

Root CA certificates

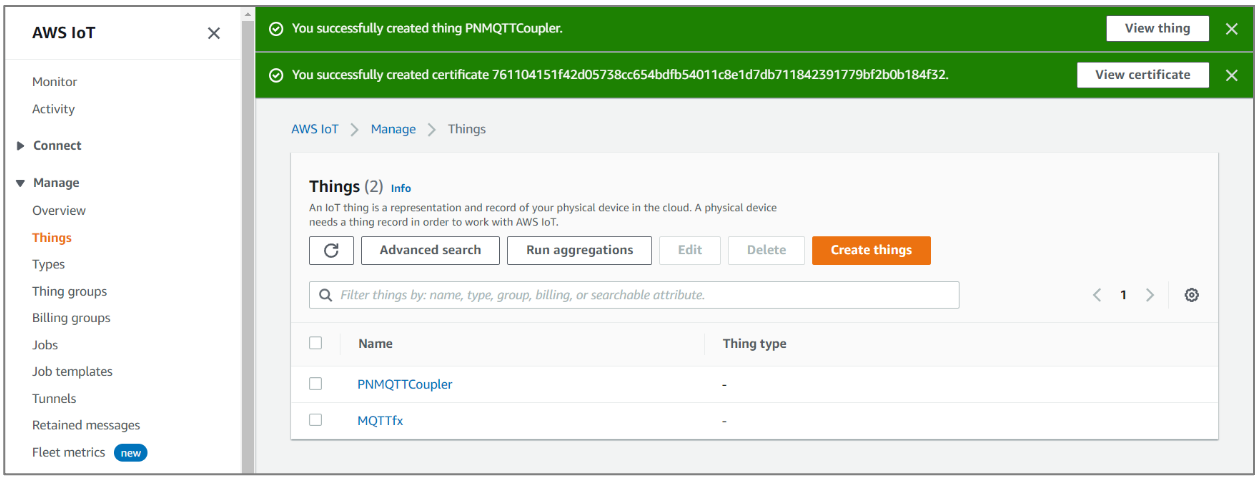
Download the root CA certificate file that corresponds to the type of data endpoint and cipher suite you're using. You can also download the root CA certificates later.

Amazon trust services endpoint
RSA 2048 bit key: Amazon Root CA 1

Amazon trust services endpoint
ECC 256 bit key: Amazon Root CA 3

If you don't see the root CA certificate that you need here, AWS IoT supports additional root CA certificates. These root CA certificates and others are available in our developer guides. Learn more

The object for the PN/MQTT Coupler is now created and we have downloaded the certificate and the key files, which we are about to import into the PN/MQTT Coupler.



You successfully created thing PNMQTTCoupler.

You successfully created certificate 761104151f42d05738cc654bdfb54011c8e1d7db711842391779bf2b0b184f32.

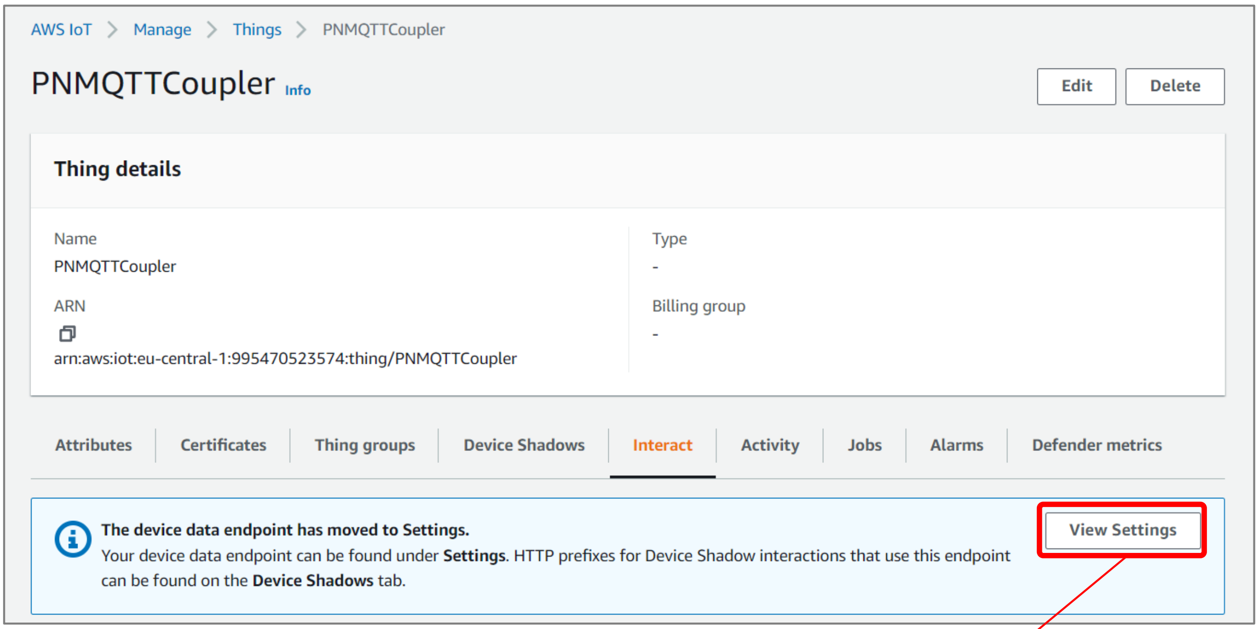
AWS IoT > Manage > Things

Things (2) Info

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

	Name	Thing type
□	PNMQTTCoupler	-
□	MQTTfx	-

Finally, we need the address of the device data endpoint. Select the object you just created and go to the "Interact" tab.



PNMQTTCoupler Info

Thing details

Name
PNMQTTCoupler

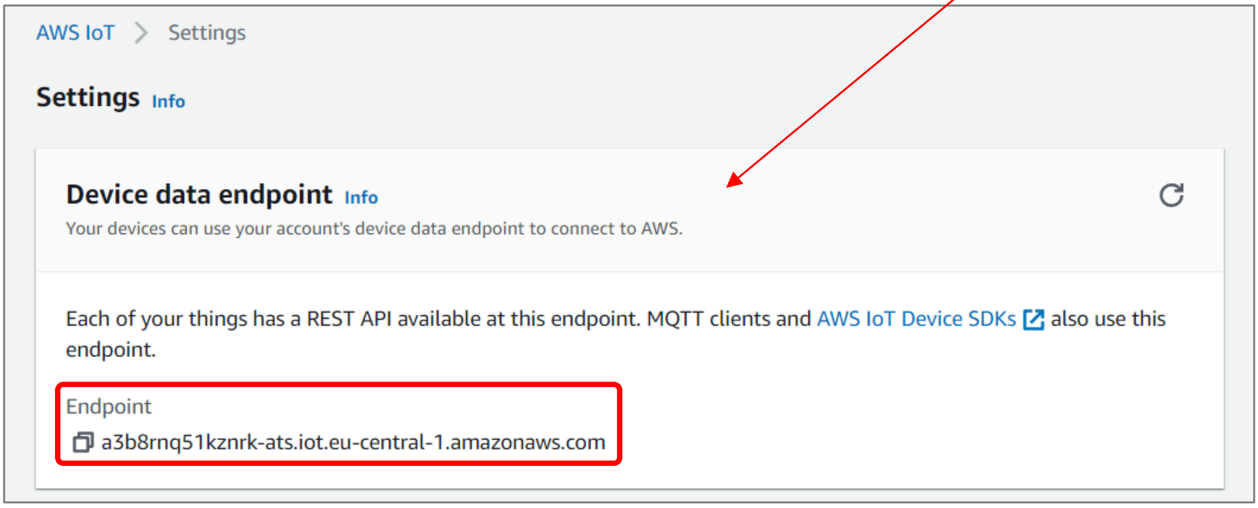
ARN
arn:aws:iot:eu-central-1:995470523574:thing/PNMQTTCoupler

Type
-

Billing group
-

The device data endpoint has moved to Settings.
Your device data endpoint can be found under **Settings**. HTTP prefixes for Device Shadow interactions that use this endpoint can be found on the **Device Shadows** tab.

With "Show settings" you get to the device data endpoint.



Settings Info

Device data endpoint Info
Your devices can use your account's device data endpoint to connect to AWS.

Each of your things has a REST API available at this endpoint. MQTT clients and AWS IoT Device SDKs also use this endpoint.

Endpoint
a3b8rnq51kznrk-ats.iot.eu-central-1.amazonaws.com

Copy the access URL "Endpoint" for this device.

17.3 Configure PN/MQTT Coupler for AWS Access

Configure the PN/MQTT coupler in the PROFINET engineering tool (e.g. TIA Portal) as described in chapter 7. When parameterizing the MQTT Topic modules, note that AWS only supports QoS '0' and '1' and that no "Retain" flag may be set for the Publisher modules!

ATTENTION AWS IoT Core has some limitations with the MQTT messages:

1. the "Retain Flag" must not be used!
2. "QoS 2" cannot be used with AWS!
3. "Keep-alive" must be between 30 and 1200 seconds!

To connect to AWS IoT Core, it is mandatory to specify a gateway and DNS server in the IP Settings.

On the PN/MQTT Coupler website, the following settings must now be made in the "MQTT" menu under "MQTT Client Settings":

AWS IoT Core currently only supports **MQTT version 3.1.1**.

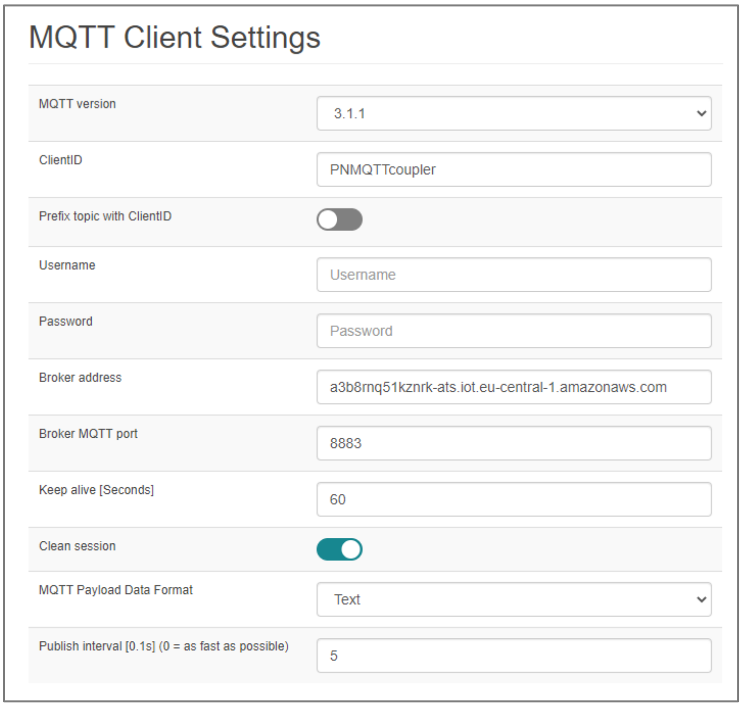
The **ClientID** can be any.

Username and **password** are not required.

At **Broker address**, the endpoint URL copied on the previous page must be pasted.

The **Broker port** must be set to 8883.

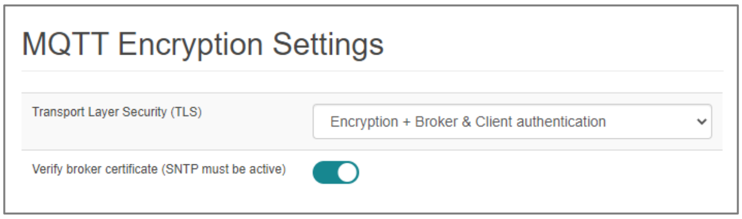
Keep alive must be set between 30 and 1200 seconds for AWS.



MQTT Client Settings

MQTT version	3.1.1
ClientID	PNMQTTcoupler
Prefix topic with ClientID	○
Username	Username
Password	Password
Broker address	a3b8rnq51kznrk-ats.iot.eu-central-1.amazonaws.com
Broker MQTT port	8883
Keep alive [Seconds]	60
Clean session	●
MQTT Payload Data Format	Text
Publish interval [0.1s] (0 = as fast as possible)	5

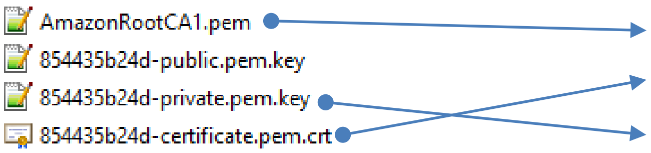
Transport Layer Security (TLS) must be set to "Encryption +Broker & Client authentication".

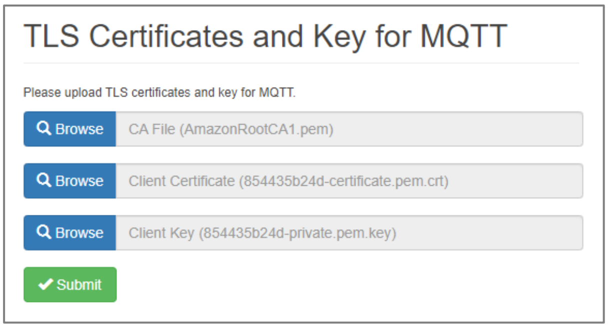


MQTT Encryption Settings

Transport Layer Security (TLS)	Encryption + Broker & Client authentication
Verify broker certificate (SNTP must be active)	●

Use the certificates downloaded on page 51 in the TLS Certificates dialog:


AmazonRootCA1.pem
854435b24d-public.pem.key
854435b24d-private.pem.key
854435b24d-certificate.pem.crt



TLS Certificates and Key for MQTT

Please upload TLS certificates and key for MQTT.

Browse
CA File (AmazonRootCA1.pem)
Browse
Client Certificate (854435b24d-certificate.pem.crt)
Browse
Client Key (854435b24d-private.pem.key)
✔ Submit

For the CA file, use the "AmazonRootCA1.pem" file.

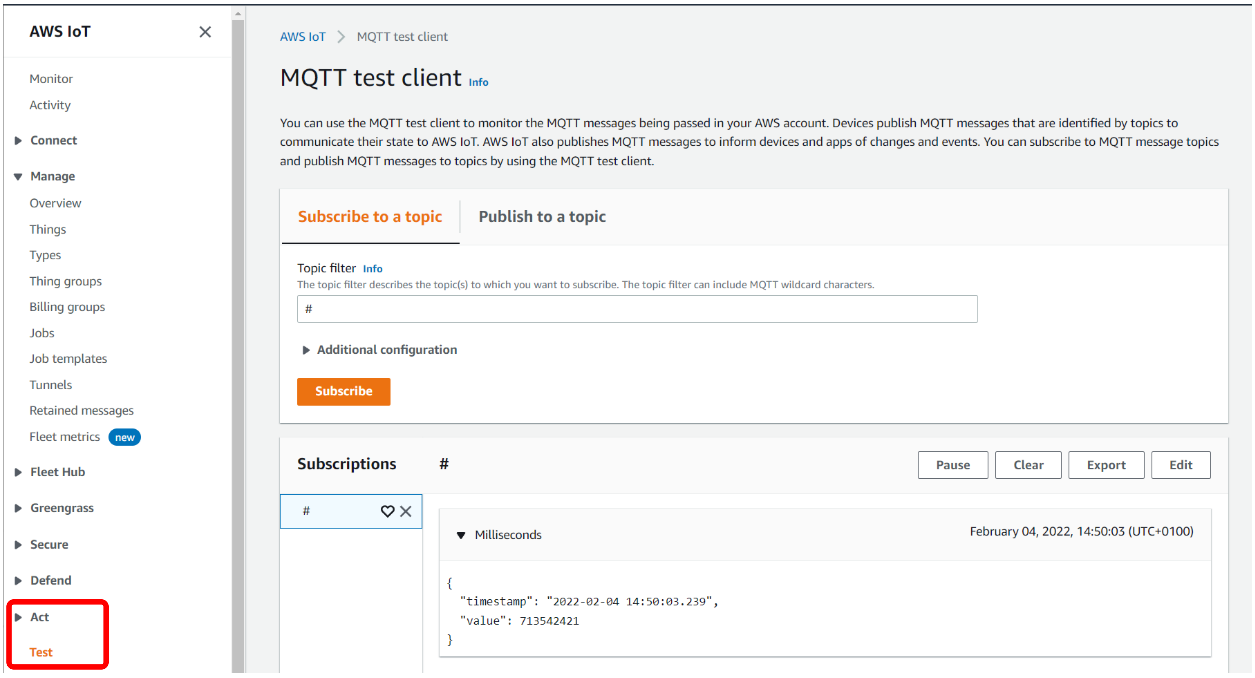
The client certificate is the "xxx-certificate.pem.crt" file, it contains both the certificate and the public key of the client.

The last thing you need is the private key of the client ("xxx-private.pem.key").

This completes the configuration and the PN/MQTT Coupler should connect to the AWS IoT Core.

17.4 Testing the MQTT Connection in AWS

To check whether the data exchange with the PN/MQTT Coupler works via MQTT, MQTT Test Client can be called in the AWS IoT Core. Select "Act/Test" from the menu on the left.



You can use the MQTT test client to monitor the MQTT messages being passed in your AWS account. Devices publish MQTT messages that are identified by topics to communicate their state to AWS IoT. AWS IoT also publishes MQTT messages to inform devices and apps of changes and events. You can subscribe to MQTT message topics and publish MQTT messages to topics by using the MQTT test client.

Topic filter Info: #

The topic filter describes the topic(s) to which you want to subscribe. The topic filter can include MQTT wildcard characters.

Subscriptions: #

February 04, 2022, 14:50:03 (UTC+0100)

```
{
  "timestamp": "2022-02-04 14:50:03.239",
  "value": 713542421
}
```

In the following dialog you can activate the display of a topic sent by the PN/MQTT Coupler under "Subscribe to topic" and send data to the PN/MQTT Coupler under "Publish to topic".

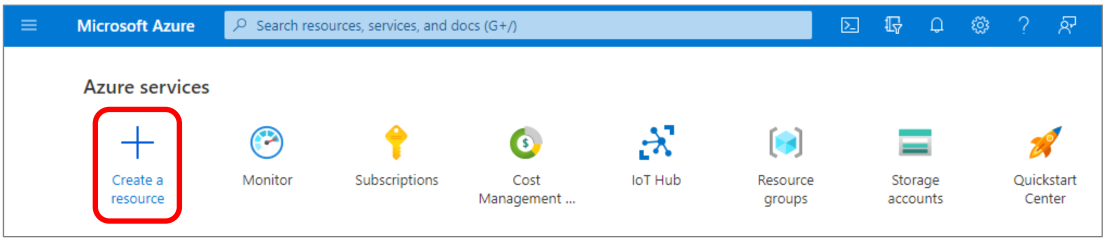
If the test is successful, the configuration is completed!

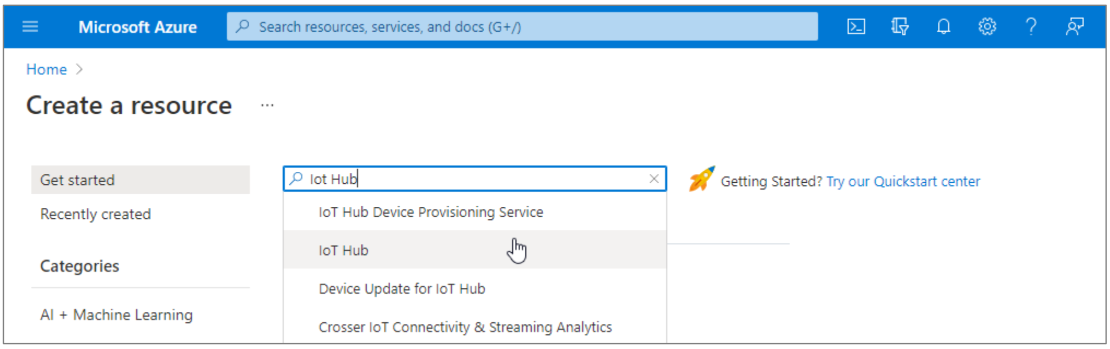
ATTENTION For the connection with amazon AWS, it is mandatory to specify a gateway and a DNS server in the IP settings. To check that the certificates are up to date, activate time synchronization via SNTP.

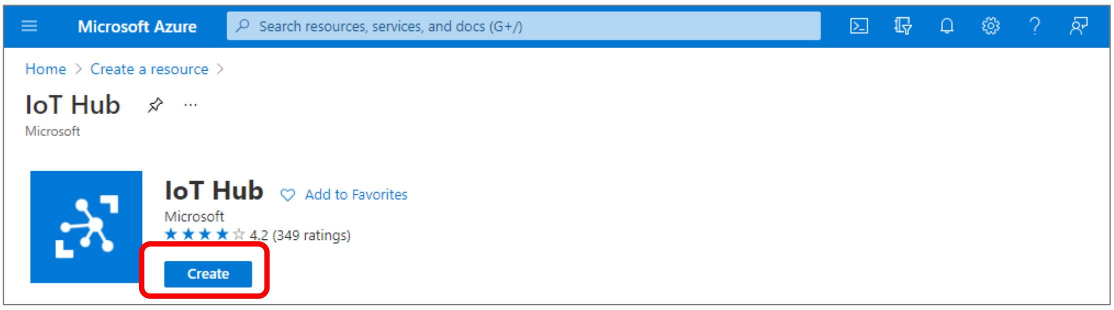
18 Application example „Microsoft Azure“

18.1 Create device in Azure

Under Microsoft Azure, an IoT Hub must first be created.




Microsoft Azure
Search resources, services, and docs (G+/)
Home
Create a resource
Get started
Recently created
Categories
AI + Machine Learning
Iot Hub
IoT Hub Device Provisioning Service
IoT Hub
Device Update for IoT Hub
Crosser IoT Connectivity & Streaming Analytics
Getting Started? Try our Quickstart center

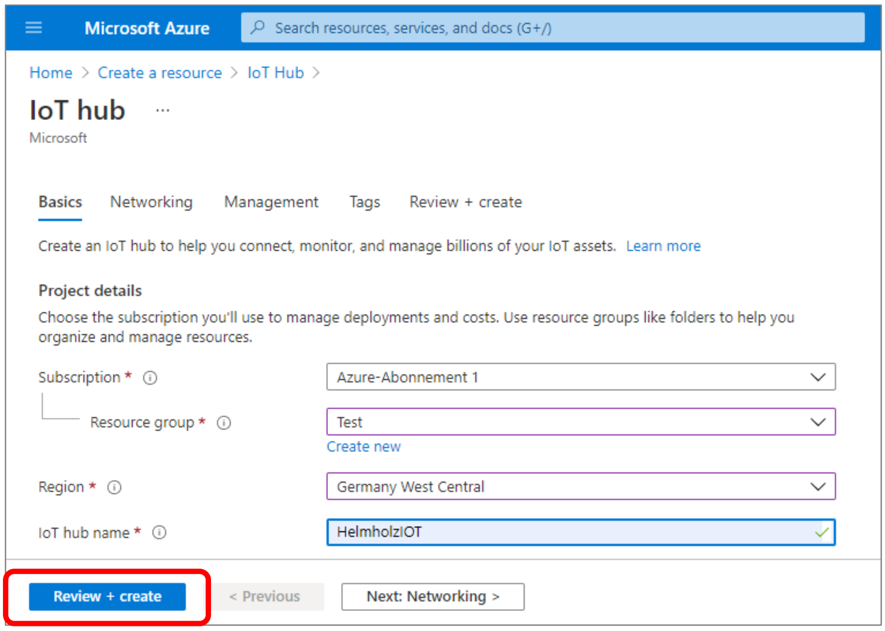


Choose an appropriate subscription. For a new AWS account, the "Free Trial" can be used if applicable.

The IoT Hub name can be chosen arbitrarily.

In the next dialog, the details can be checked again and "Create" starts the deployment of the IoT Hub component in Azure.

Deployment may take a few minutes.



Create an IoT hub to help you connect, monitor, and manage billions of your IoT assets.

Project details

Choose the subscription you'll use to manage deployments and costs. Use resource groups like folders to help you organize and manage resources.

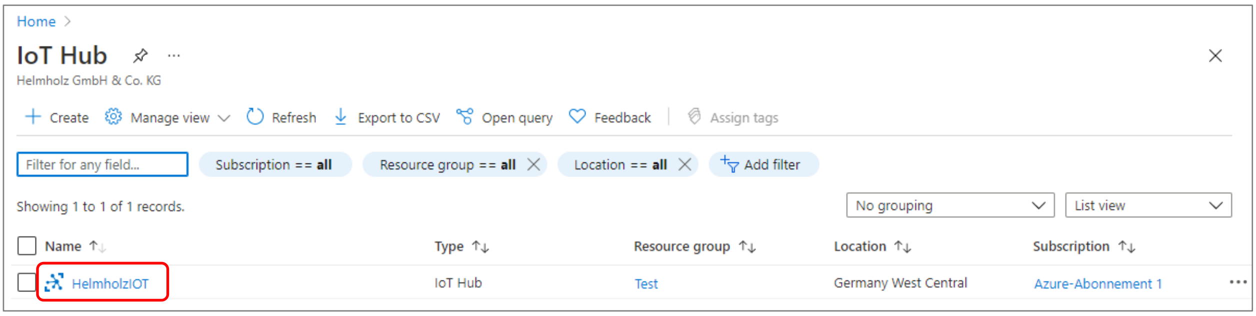
Subscription *: Azure-Abonnement 1

Resource group *: Test

Region *: Germany West Central

IoT hub name *: HelmholzIOT

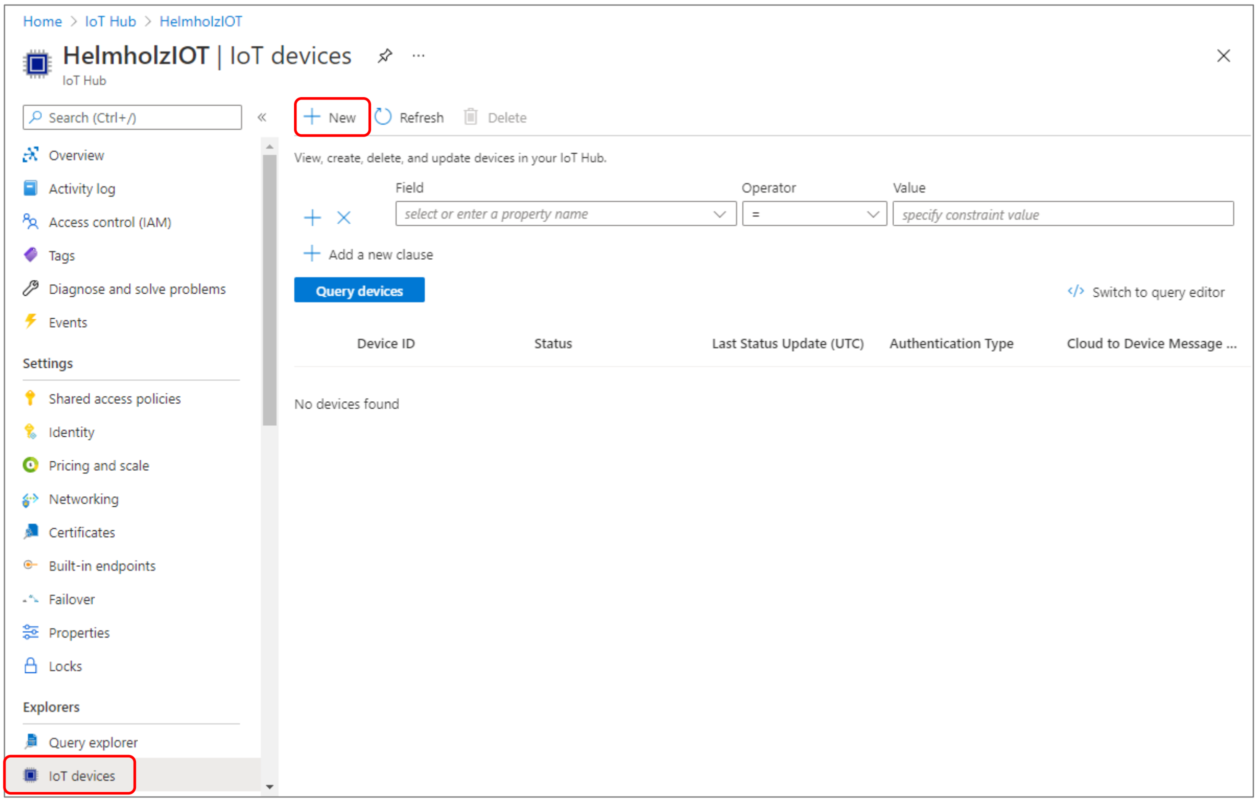
Then select the newly created IoT Hub.



Showing 1 to 1 of 1 records.

Name	Type	Resource group	Location	Subscription
HelmholzIOT	IoT Hub	Test	Germany West Central	Azure-Abonnement 1

In the menu tree, select "IoT devices" at the bottom left and create a new device with "New" at the top.



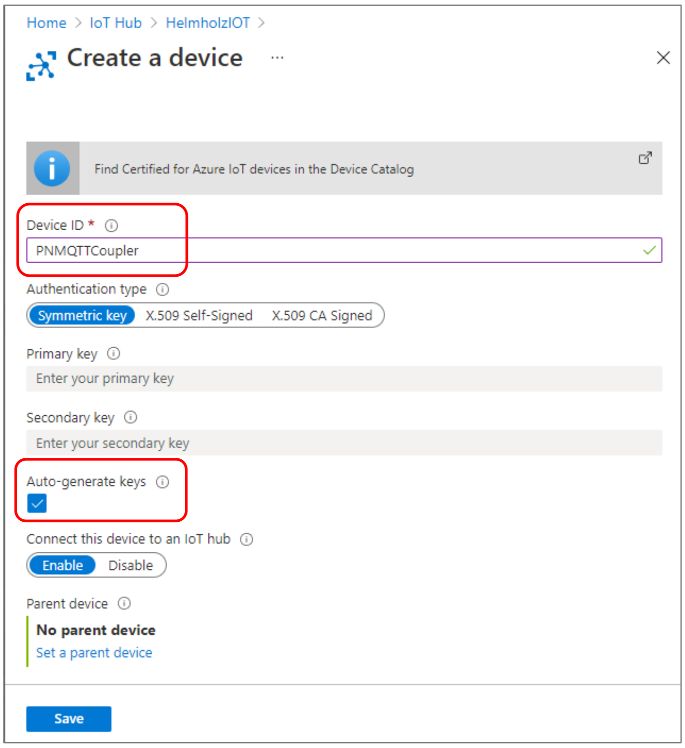
View, create, delete, and update devices in your IoT Hub.

Device ID	Status	Last Status Update (UTC)	Authentication Type	Cloud to Device Message ...

No devices found

In the following dialog you can give the device a device ID.

The other settings can be used unchanged, important are the options" Symmetric key" and Generate key automatically.



Create a device

Find Certified for Azure IoT devices in the Device Catalog

Device ID * : PNMQTTCoupler

Authentication type: [✓ Symmetric key] X.509 Self-Signed X.509 CA Signed

Primary key: Enter your primary key

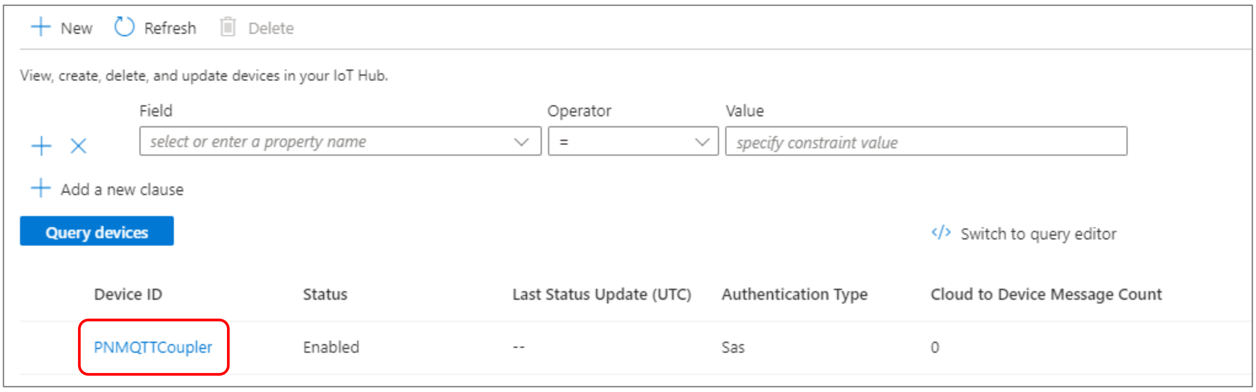
Secondary key: Enter your secondary key

Auto-generate keys: ☑

Connect this device to an IoT hub: [✓ Enable] Disable

Parent device: **No parent device**

Set a parent device

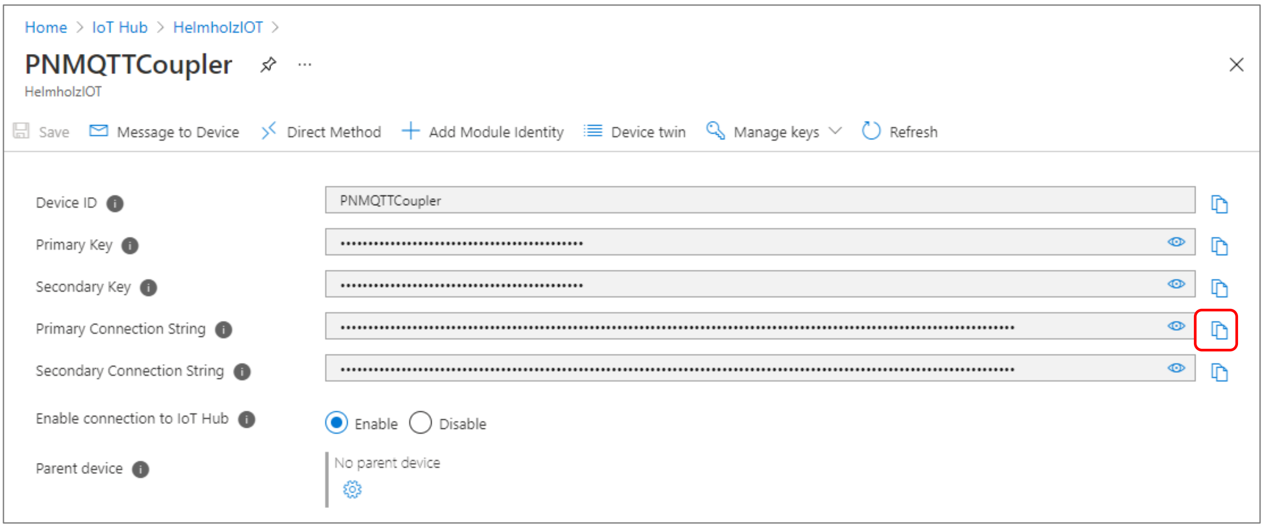


View, create, delete, and update devices in your IoT Hub.

Field	Operator	Value
select or enter a property name	=	specify constraint value

Device ID	Status	Last Status Update (UTC)	Authentication Type	Cloud to Device Message Count
PNMQTTCoupler	Enabled	--	Sas	0

Select the device.



PNMQTTCoupler

HelmholzIOT

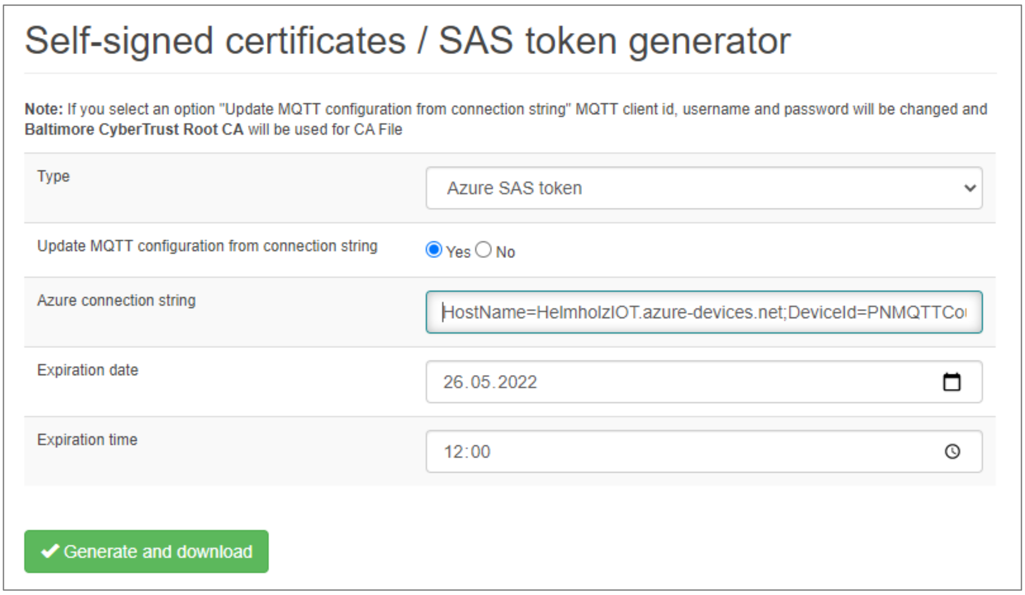
Device ID	PNMQTTCoupler
Primary Key	••
Secondary Key	••
Primary Connection String	••
Secondary Connection String	••
Enable connection to IoT Hub	● Enable ○ Disable
Parent device	No parent device

Copy the "Primary Connection String" to the clipboard.

18.2 Configure PN/MQTT Coupler for Azure

The PN/MQTT coupler must already be configured via PROFINET and have an IP address on the MQTT network side and be able to establish a connection to the Internet (gateway and DNS server are available).

First, an "Azure SAS Token" must be created for the coupler in the PN/MQTT Coupler in the "MQTT" menu under "MQTT Encryption".



Self-signed certificates / SAS token generator

Note: If you select an option "Update MQTT configuration from connection string" MQTT client id, username and password will be changed and **Baltimore CyberTrust Root CA** will be used for CA File

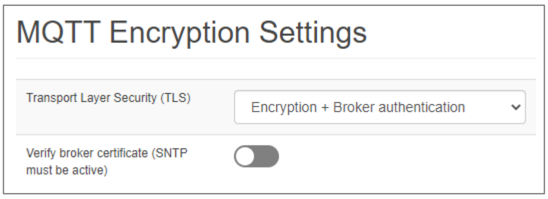
Type	Azure SAS token
Update MQTT configuration from connection string	● Yes ○ No
Azure connection string	HostName=HelmholzIOT.azure-devices.net;DeviceId=PNMQTTCo
Expiration date	26.05.2022
Expiration time	12:00

The "Primary connection string" from the clipboard (see previous page) must be entered as the "Azure connection string". As "Expiration date" and "Expiration time" a time in the future must be entered.

With "Update MQTT configuration from connection string", the connection settings are automatically transferred to the "MQTT Settings". This includes "Username", "Password" and the "Broker address".

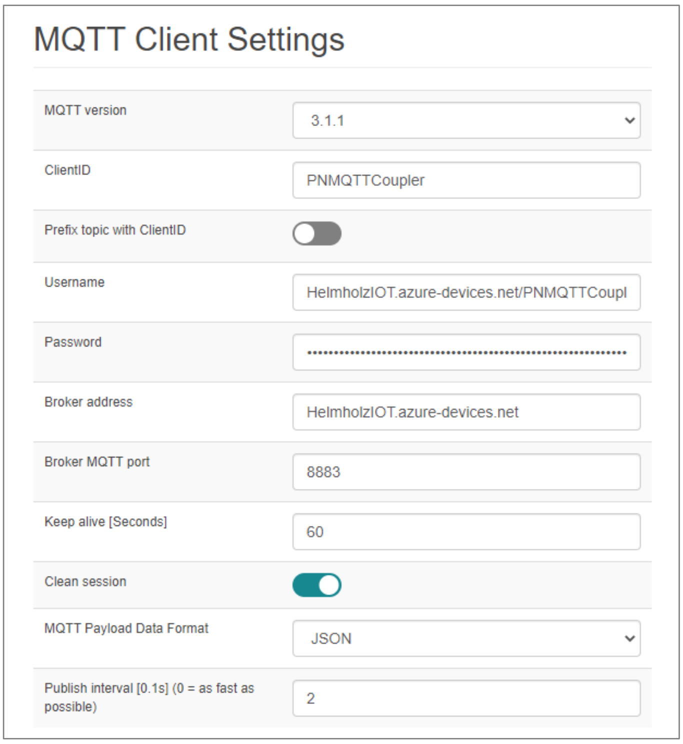
In the "MQTT Client Settings", check the remaining settings as shown on the right. The broker port must be set to 8883.

The MQTT Encryption must be set to "Encryption + Broker authentication".



MQTT Encryption Settings

Transport Layer Security (TLS)	Encryption + Broker authentication
Verify broker certificate (SNTP must be active)	off

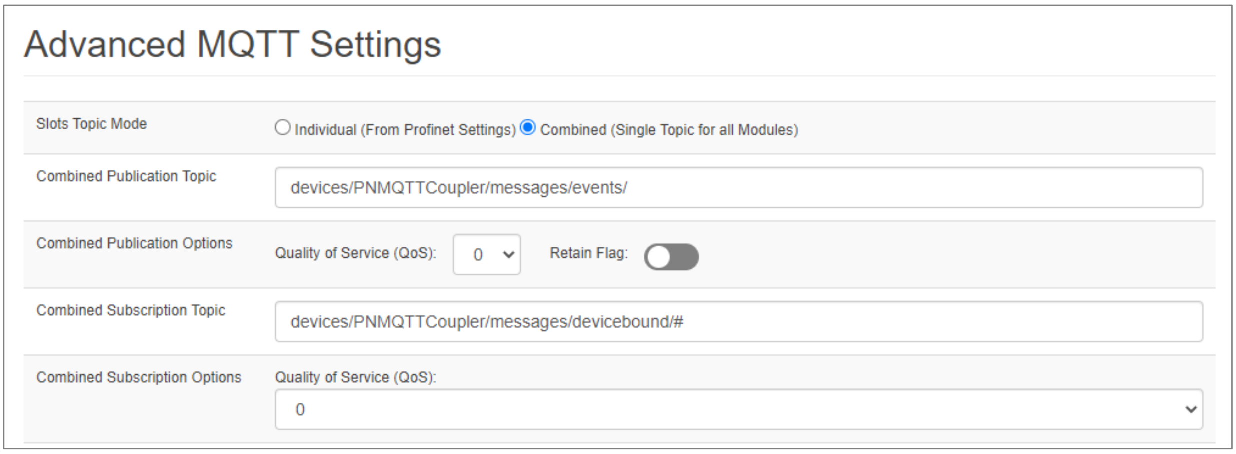


MQTT Client Settings

MQTT version	3.1.1
ClientID	PNMQTTCoupler
Prefix topic with ClientID	off
Username	HelmholzIOT.azure-devices.net/PNMQTTCoupl
Password	••
Broker address	HelmholzIOT.azure-devices.net
Broker MQTT port	8883
Keep alive [Seconds]	60
Clean session	on
MQTT Payload Data Format	JSON
Publish interval [0.1s] (0 = as fast as possible)	2

Since Microsoft Azure can only receive one central topic from each device, all configured values from the PLC must be sent together in a combined topic.

Select "MQTT / Advanced MQTT settings" and check whether the "Combined Topic mode" is activated.



Advanced MQTT Settings

Slots Topic Mode	○ Individual (From Profinet Settings) ● Combined (Single Topic for all Modules)
Combined Publication Topic	devices/PNMQTTCoupler/messages/events/
Combined Publication Options	Quality of Service (QoS): 0 Retain Flag:
Combined Subscription Topic	devices/PNMQTTCoupler/messages/devicebound/#
Combined Subscription Options	Quality of Service (QoS): 0

The “Combined Publication Topic” must be set to the following format for Microsoft Azure:

```
devices/<device-ID>/messages/events/
```

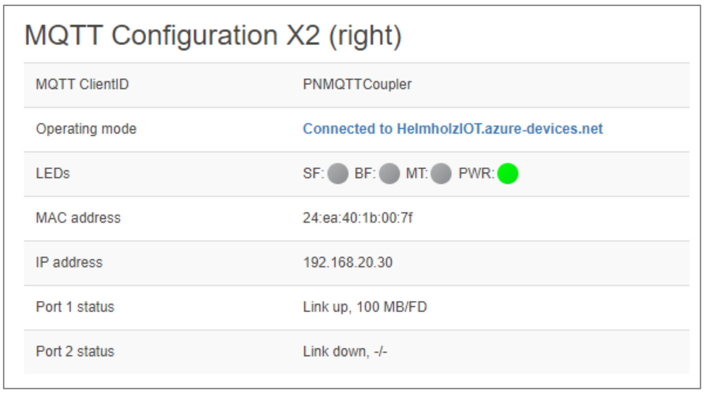
The “Combine Subscription Topic” must be set to the following format for Microsoft Azure:

```
devices/<device-ID>/messages/devicebound/#
```

The <device-ID> of the device created in Azure must be specified correctly here, e.g.:

```
devices/PNMQTTCoupler/messages/events/
```

If all settings are correct, the PN/MQTT Coupler should establish the connection to Microsoft Azure by itself.



MQTT Configuration X2 (right)

MQTT ClientID	PNMQTTCoupler
Operating mode	Connected to HelmholzIOT.azure-devices.net
LEDs	SF: ● BF: ● MT: ● PWR: ●
MAC address	24:ea:40:1b:00:7f
IP address	192.168.20.30
Port 1 status	Link up, 100 MB/FD
Port 2 status	Link down, -/-

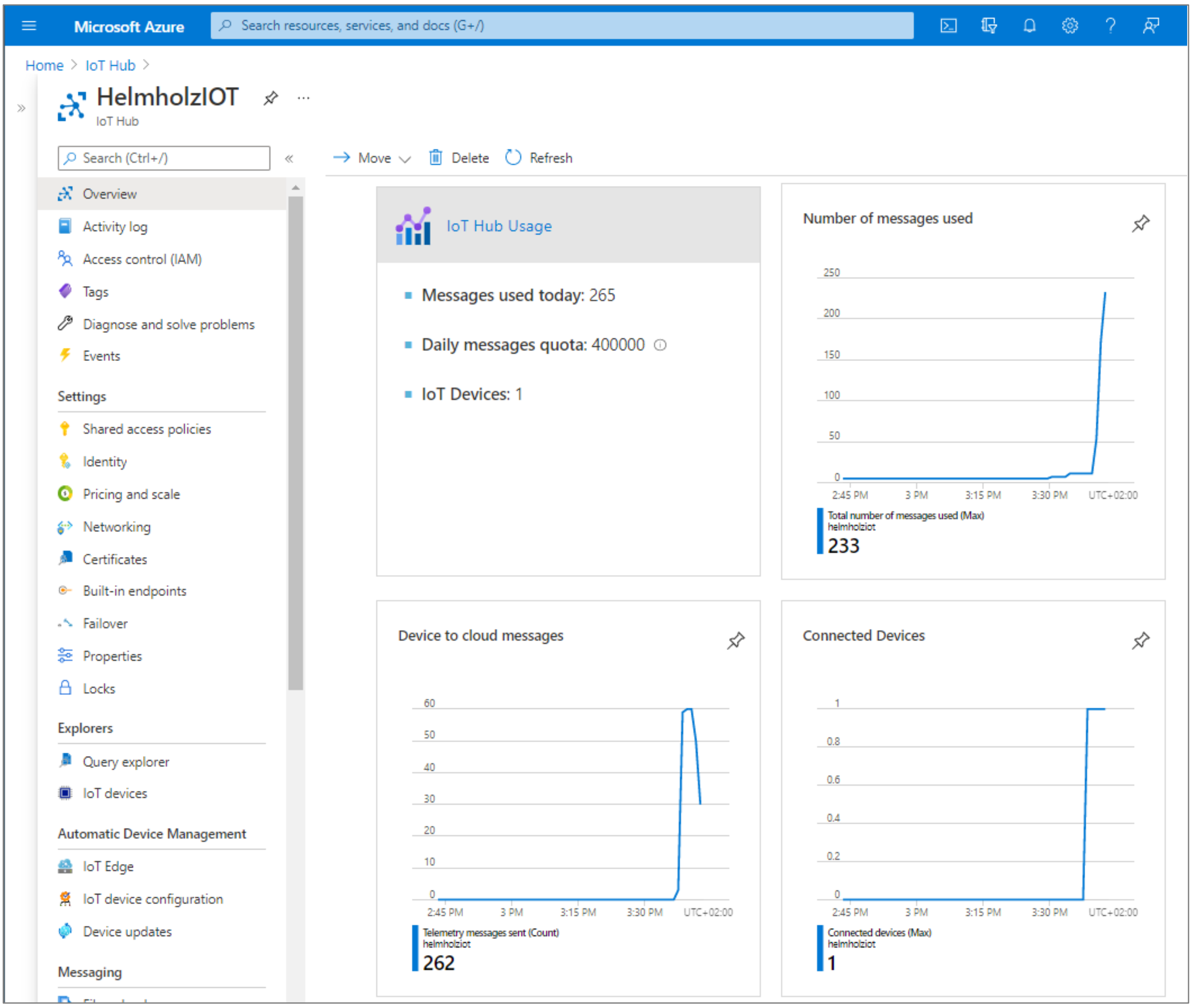
ACHTUNG Microsoft Azure has some restrictions:

1. the "Retain flag" must not be used!
2. "QoS = 2" cannot be used with Azure!
3. keep alive is limited to a maximum of 19 min (1140 seconds).
4. "Communication enable" & "Communication stopped" messages must not be used.

For the connection with Microsoft Azure, it is mandatory to specify a gateway and a DNS server in the IP settings.
Time synchronization via SNTP must be activated to check whether the certificates are up-to-date.

18.3 Checking the data transfer in Microsoft Azure

Whether there is an active connection of the PN/MQTT Coupler with the IoT Hub can be seen in the IoT Hub overview under "IoT Hub usage". The active connections and the number of messages used are displayed there.



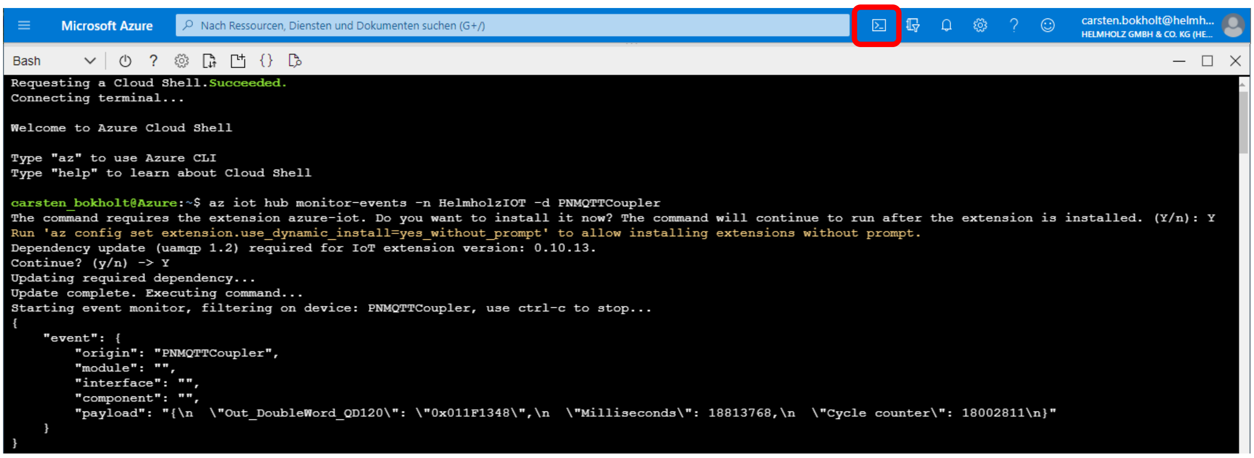
- Messages used today: 265
- Daily messages quota: 400000
- IoT Devices: 1

Additionally, you can view the received messages in the Azure Console. The following command starts the MQTT event monitoring:

```
az iot hub monitor-events -n <Hub-Name> -d <device-ID>
```

example:

```
az iot hub monitor-events -n HelmholzIOT -d PNMQTTCoupler
```


Microsoft Azure
Nach Ressourcen, Diensten und Dokumenten suchen (G+/)
carsten.bokholt@helmh...
HELMHOLZ GMBH & CO. KG (HE...
Bash

```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

carsten_bokholt@Azure:~$ az iot hub monitor-events -n HelmholzIOT -d PNMQTTCoupler
The command requires the extension azure-iot. Do you want to install it now? The command will continue to run after the extension is installed. (Y/n): Y
Run 'az config set extension.use_dynamic_install=yes_without_prompt' to allow installing extensions without prompt.
Dependency update (uamqp 1.2) required for IoT extension version: 0.10.13.
Continue? (y/n) -> Y
Updating required dependency...
Update complete. Executing command...
Starting event monitor, filtering on device: PNMQTTCoupler, use ctrl-c to stop...
{
    "event": {
        "origin": "PNMQTTCoupler",
        "module": "",
        "interface": "",
        "component": "",
        "payload": "{\n  \"Out_DoubleWord_QD120\": \"0x011F1348\",\n  \"Milliseconds\": 18813768,\n  \"Cycle counter\": 18002811\n}"
    }
}
```

19 Technical data

Order no.	**700-162-3MQ02**
Article designation	PN/MQTT Coupler
PROFINET interface (X1)	
Connection	2x RJ45, integrated switch
Protocol	PROFINET IO Device as defined in IEC 61158-6-10
Transmission rate	100 Mbit/s full duplex
I/O image size	Up to 1024 Byte of input and output data
Number of configurable slots	100
Features	PROFINET Conformance Class B, media redundancy (MRP-Client), automatic addressing, Topology detection (LLDP, DCP), diagnosis alarms
MQTT interface (X2)	
Connection	2x RJ45, integrated switch
Protocol	MQTT V3.1.1 & V5
Transmission rate	10/100 Mbit/s, full-/half duplex
Status indicator	9 LEDs function status, 8 LEDs Ethernet-status
Voltage supply	DC 24 V (18 - 28 V DC)
Current draw	max. 210mA
Power dissipation	max. 5 W
Dimensions (D x W x H)	32,5 x 58,5 x 76 mm (without power supply connector)
Weight	approx. 135 g
Certifications	PROFINET Conformance Class B
Ambient conditions	
Protection rating	IP 20
Ambient temperature	0° C to 60° C
Transport and storage temperature	-20° C to 80° C
Relative humidity	95% non-condensing
Mounting position	any
Noise immunity	DIN EN 61000-6-2 “EMC Immunity”
Interference emission	DIN EN 61000-6-4 “EMC Emission”
Vibration and shock resistance	DIN EN 60068-2-6:2008 “Vibration” DIN EN 60068-2-27:2010 “Shock”