

Application Note

Synchronised Operation of Smart Directors via MQTT

MICROSENS GmbH & Co. KG
Kueferstr. 16
59067 Hamm/Germany

Tel. +49 2381 9452-0
FAX +49 2381 9452-100
E-Mail info@microsens.de
Web www.microsens.de

Summary

This Application Note helps to configure an MQTT broker within a MICROSENS Smart Business infrastructure. It explains the setup of the MQTT broker for the use of basic MQTT features.

The content of the document builds on information provided with the Quick Start Guide that is shipped together with MICROSENS Smart Building devices. Additionally, familiarity with the use of MICROSENS Smart Building Manager is required to take advantage out of this Application Note.

For further information about using MICROSENS Smart Building Manager please refer to the respective user manual.

For further information about using the Web Manager for Smart Office Configuration of MICROSENS Smart Building devices please refer to the chapter "Smart Office" in [1] (see page 2).

Glossary

Terms, acronyms and abbreviations used in the document.

Term	Description
Local Broker	MQTT broker which is running on the same device as the publishing MQTT client
MQTT	Message Queuing Telemetry Transport
MQTT broker	The central MQTT server of the Smart Building network
MQTT client	Smart Building devices connected to the MQTT broker
Publisher	MQTT client that sends messages of a specific topic to the MQTT broker
QoS	Quality of Service
Subscriber	MQTT client that has subscribed to messages of a specific topic
Topic	Hierachically organised category of an MQTT message

Bibliography

[1] Product Manual Firmware Generation 6


This manual is included in each software archive containing Firmware G6. It can also be downloaded from the link "Documentation" in the firmwares' Web Manager navigation bar.

Table of Contents

- SUMMARY..... 2**
- GLOSSARY..... 2**
- BIBLIOGRAPHY 2**
- TABLE OF CONTENTS..... 3**
- TYPOGRAPHICAL CONVENTIONS..... 4**
- INFORMATION AVAILABLE FROM THE MICROSENS WEBSITE..... 4**
- 1 INTRODUCTION TO MQTT 5**
 - 1.1 Publish/Subscribe Concept 5
 - 1.2 MQTT Broker 5
 - 1.3 MQTT Clients 5
 - 1.4 MQTT Publish Message Format..... 5
 - 1.5 MQTT Subscribe Message Format 7
 - 1.6 MQTT Unsubscribe Message Format..... 7
 - 1.7 MQTT Message Sequence Example 8
- 2 USING MQTT FOR DECENTRALISED ROOM ACCESS 10**
 - 2.1 Prerequisites for Configuring MQTT.....11
 - 2.2 Setup the Web GUI for Central Functions11
 - 2.3 Configuring MQTT Broker and MQTT Clients.....13
 - 2.3.1 Using Variables.....14
 - 2.3.2 Setting up the Local MQTT Broker14
 - 2.3.3 Setting up the Local MQTT Client18
 - 2.3.4 Setting up the Remote MQTT Clients19
- DISCLAIMER..... 22**

Typographical Conventions

The following typographical elements are used in this document:

Typographical Elements	Explanation
•	List element, 1 st order
◦	List element, 2 nd order
www.microsens.de	Hyperlink to a website or email address
Note:	A note tags an important fact
□	Work step
<...>	Placeholder for a real value. Replace <IP Address> with e.g. 192.168.1.2.
Visualisation	A string that appears in the Web Manager
Command	A string to enter in the Command Line Interface
» Output	A string output by the Command Line Interface
	Work step(s) in the Web Manager (GUI)

Information available from the MICROSENS Website

Registered users can find the latest firmware and MICROSENS Smart Director app versions as well as further information on our web site:

- Registration: www.microsens.de > Partner-Login > Follow the link 'Please register here' > Fill in the online registration form and submit it
 - You will receive an email from MICROSENS with a user name and a password
- Login: www.microsens.de > Partner-Login > Enter user name and password > Click the 'Login' button
 - Firmware images: Navigate to the device and select the tab 'Services'
 - For further information select one of the other tabs

Note:

Make sure the browser allows the execution of scripts.

1 Introduction to MQTT

MQTT is a TCP/IP-based light-weight messaging protocol that allows multiple Smart Building devices (clients) to communicate via an MQTT server ("Broker"). The MQTT broker manages the communication between all connected devices and enables (due to the slim MQTT protocol) even constrained devices sending and receiving data in networks with limited bandwidth.

Because of relaying the device communication through the MQTT broker it is easy to extend the Smart Building network with new devices without touching the network infrastructure. It is not even necessary that new devices have to be compatible with existing devices since they only communicate with the MQTT broker.

1.1 Publish/Subscribe Concept

MQTT is based on a "Publish/Subscribe" concept: A device can publish a message of a specific type ("Topic") to the network. Respective devices that are subscribed to the MQTT broker for this topic automatically receive and interpret the message. This message can either contain commands or data. Both publisher and subscriber do not need to know each other being part of the network ("space decoupling").

1.2 MQTT Broker

The MQTT broker performs various tasks:

1. Managing connected MQTT clients (i.e. credentials, addresses, subscriptions, client's "last will" etc.).
2. Managing topics and messages (i.e. receiving all messages from the connected clients, filtering the messages and forwarding them to the respective subscribed clients). Forwarding messages happens asynchronously which means that publishing and subscribing clients do not necessarily have to be active at the same time ("time decoupling").
3. Ensuring data security and Quality of Service (QoS).

1.3 MQTT Clients

Every Smart Building device connected to a MQTT broker acts as MQTT client, clearly detected by its specific client ID. This device can be a sensor (e.g. for temperature, motion detection, humidity, button clicks etc.) or an actuator (e.g. power switch, motor, fan, blind, etc.).

MICROSENS Smart Building uses MQTT to link Smart Directors as MQTT clients for remote access

An MQTT client can hold both roles as publisher and subscriber for specific topics. The MQTT protocol does not supervise whether a subscribed topic fits a client's feature. The client itself has to make sure to publish and subscribe suitable topics.

1.4 MQTT Publish Message Format

An MQTT message, that is sent from a publishing MQTT client, consists of six parts:

1. Packet ID:

The packet ID is an identifier that ensures the unique message flow between both MQTT

client and broker. Depending on the QoS level either client or broker are responsible for setting this ID automatically.

2. **Topic:**

A topic can be understood as the category of a message. Topics are structured hierarchically (with a forward slash as delimiter between levels), comparable to a file system structure (e.g. "Building/Floor1/Room1/CeilingLight").

Topics are defined by the user, where a user friendly self descriptive naming convention mirrors the Smart Building infrastructure.

Topic names are case sensitive (".../CeilingLight" differs from ".../ceilinglight") and must contain at least one character.

Note:

It is possible to use every UTF-8 character (besides "\$" since this character is used by the broker for internal statistics).

The use of the following wildcards is possible:

+: This character replaces a single level in a topic.

Example: "Building/Floor1/+/Temperature"

This topic addresses the "Temperature" related messages for all rooms on "Floor1".

#: This character replaces multiple levels in a topic.

Example: "Building/Floor1/#"

This topic addresses all occurring messages on "Floor1".

Note:

The use of wildcards is allowed when using microScript to register topics.

It is not allowed using the MQTT mapping table and matching multiple topics to only one component is not appropriate (e.g. matching a sensor to a topic containing multiple rooms).

3. **QoS:**

The Quality of Service value determines the level of transfer reliability for the message reaching the subscriber.

For MQTT there are three possible QoS levels:

0: "at most once"

- no guarantee for message delivery
- no acknowledgement of receiving the message by MQTT broker
- no storing or re-transmitting the message by MQTT publisher
- Packet ID automatically set to "0"

1: "at least once"

- guarantee for delivering the message successfully at least one time to the broker
- storing and re-transmitting the message unless acknowledgement by the broker
- acknowledgement contains the unique packet ID only, so the publisher can assign message and acknowledgement

2: "exactly once"

- guarantee for delivering every message exactly one time to the broker
- publisher and broker using a four-part handshake for sending and acknowledgement

- acknowledgement messages between publisher and broker contain the packet ID only to assign message and acknowledgements

4. **Retain Flag:**

This flag determines whether the broker will save this message as the last valid sample for this specific topic. In case a new MQTT client subscribes for this topic the broker transmits this message to the subscriber.

5. **Payload:**

The content of the MQTT message is the payload that suits the specific topic. It may contain data in form of text, images or every other binary data, formatted in any encodings and encryptions.

The user is responsible for assigning the clients payload to the chosen topic (e.g., a temperature sensor should publish the measured temperature with a respective topic "Room/Thermostat", not "Room/Switch").

6. **DUP flag:**

This flag indicates that the message was published before but the publisher did not receive an acknowledgement yet for the original message.

1.5 MQTT Subscribe Message Format

An MQTT message, that is sent from a subscribing MQTT client, consists of two parts:

1. **Packet ID:**

The packet ID is an identifier that ensures the unique message flow between both MQTT client and broker. Client or broker are responsible for setting this ID automatically.

2. **Subscription Set:**

A client can register to multiple subscriptions in one MQTT subscribe message. Every subscription set contains two elements:

QoS: Defines the QoS level for a subscribed topic. With overlapping topic subscriptions the broker transmits the message with the highest QoS level.

Topic: Contains the topic the client wants to subscribe to. The use of wildcards is possible.

1.6 MQTT Unsubscribe Message Format

An MQTT message, that unsubscribes an MQTT client from one or multiple topics, consists of two parts:

1. **Packet ID:**

The packet ID is an identifier that ensures the unique message flow between both MQTT client and broker. Client or broker are responsible for setting this ID automatically.

2. **Topic Set:**

A client can resign from one or multiple subscriptions in one MQTT unsubscribe message. A topic set contains every topic the client wants to unsubscribe from.

1.7 MQTT Message Sequence Example

The following example shows how MQTT clients and MQTT broker communicate with each other, assuming a QoS level of "1":

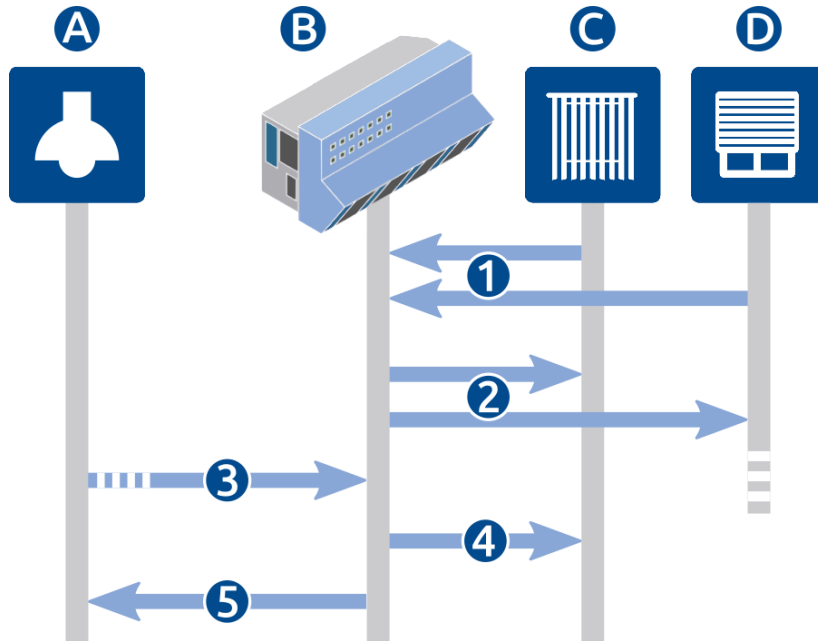


Fig. 1: MQTT Message Sequence Example

A	Publishing MQTT client containing a temperature sensor.
B	MICROSENS switch or CLC with Smart Director and MQTT broker enabled.
C	Heat valve acting as MQTT client.
D	Blinds control acting as MQTT client.
1	Both MQTT clients (C , D) send a "SUBSCRIBE" message to the broker (B). The heat valve (C) subscribes to the topic "Building/Floor1/Room1/Temperature". The blinds control (D) subscribes to the topic "Building/Floor1/Room1/Brightness".
2	The MQTT broker confirms these subscription by sending a "SUBACK" acknowledgement message to both clients.

<p>3</p>	<p>The temperature sensor (A) publishes the following MQTT message:</p> <pre>packetID: 4321 topicName: "Building/Floor1/Room1/Temperature" qos: 1 retainFlag: true payload: "25.2C" dupFlag: false</pre> <p>As long as the client does not get an acknowledgement message from the broker (B) he repeats sending the message with adjusted "dupFlag: true".</p>
<p>4</p>	<p>The MQTT broker (B) transmits the message to the client that has subscribed to the specific topic.</p> <p>In this case the heat valve (C) receives the temperature value.</p> <p>The blinds control (D) has subscribed to another topic and therefore does not receive the temperature value.</p>
<p>5</p>	<p>According to QoS level 1 the MQTT broker (B) acknowledges the publish message with a "PUBACK" message to the client (A).</p> <p>Due to "retainFlag: true" the broker stores the temperature value for new clients subscribing to this topic at a later time.</p>

2 Using MQTT for Decentralised Room Access

One Smart Director is destined to control one zone (i.e. room, floor, etc.). Therefore, the respective interface is located "on site".

Nevertheless, sometimes it is necessary to control more than one Smart Director (and therefore more than one zone) via one single interface. Every local zone interface should operate independently unless the "central" interface takes over control. The local interfaces anyway should be able to overwrite the settings performed by the "central" interface.

To achieve this all interfaces will work as MQTT clients while the MQTT broker is enabled on one G6 switch or CLC. The following steps will describe how to setup the Smart Building configuration for decentralised access to remote Smart Directors.

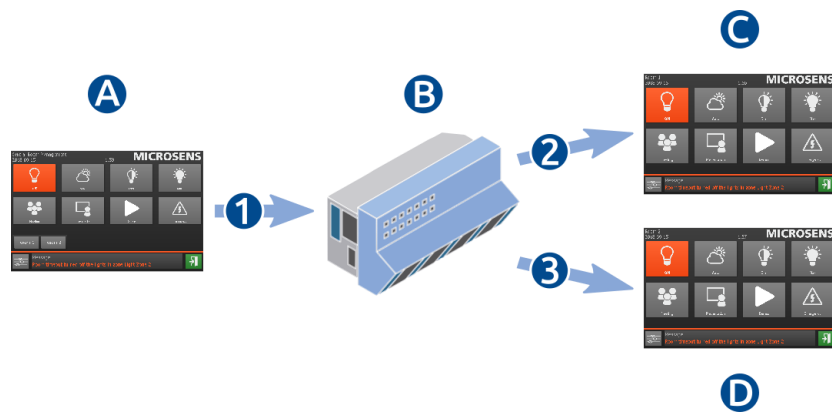


Fig. 2: Using MQTT with Smart Directors

A	Web GUI of central Smart Director, acting as MQTT client to access the remote Smart Directors in room 1 and room 2.
B	MICROSENS G6 switch or CLC with MQTT broker enabled. Note: Central Smart Director (MQTT client, A) and MQTT broker are running on the same device.
C	Web GUI of remote Smart Director, which is acting as MQTT client in room 1, subscribed to the respective topic.
D	Web GUI of Remote Smart Director, which is acting as MQTT client in room 2, subscribed to the respective topic.
1	On the web GUI of the central Smart Director (A) one of the remote access buttons is hit. The client sends a "PUBLISH" message to the broker (B), containing the topic related to the specific room and the respective button value.
2	If the message contains the topic related to room 1, the broker (B) forwards the publish message to the remote Smart Director in room 1 (C).
3	If the message contains the topic related to room 2, the broker (B) forwards the publish message to the remote Smart Director in room 2 (D).

2.1 Prerequisites for Configuring MQTT

In order to be able to configure the Smart Office components, the following basic measures must be put into place:

- Administrator access rights to the specific Smart Office device are a must-have.
- All Smart Office components have to be properly connected, installed and accessible via MICROSENS NMP, MICROSENS SBM, IPConfig Tool and MICROSENS Smart Director.
- All affected switches and CLCs must have installed the latest G6 firmware.

Note:

For more information about the respective administrator credentials and on how to access the device please refer to the device's quick start guide.

Note:

For more information about how to use MICROSENS NMP, MICROSENS SBM and IPConfig Tool please refer to the respective user manual.

The following descriptions assume you are using the Web Manager and logged in into the MICROSENS G6 switch, from where you want to configure and manage the Smart Office infrastructure.

2.2 Setup the Web GUI for Central Functions

Open the web browser of the central Smart Director device and enter the following URL:

`https://<G6 switch IP address>/gui/SmartOffice.html`

After you have successfully logged in into the Smart Engine the web GUI should start with the following default browser dialogue:



Fig. 3: Web GUI of Smart Engine

Every element of the web GUI is customisable via the Web Manager of the hosting G6 switch. For changing the scenes of remote Smart Directors the Smart Engine's web GUI of the local device should contain two additional buttons which are linked to the respective Smart Directors of the remote devices. The web GUI layout will look like the following image:



Fig. 4: Adapted Web GUI of Smart Engine

The following steps will add two buttons to the web GUI and link them to the Smart Directors of MICROSENS devices in room 1 and room 2.

Note:

The steps are illustrated using the Web Manager. The titles of the individual sections comply with the respective path names in the CLI.

Using the Web Manager on the central G6 switch or CLC:

- Open the Web Manager of the respective switch or CLC.
- Select the **Scripting** screen, then select the tab **SmartDirector**.

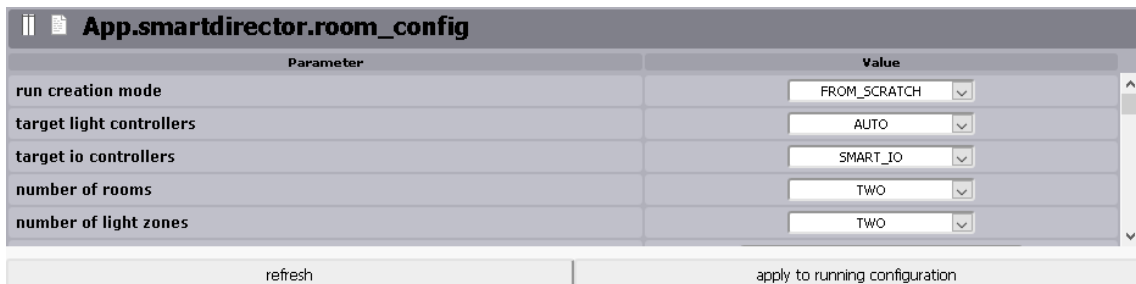


Fig. 5: Web Manager - Scripting - SmartDirector – Room Config

- In the section **App.smartdirector.room_config** change the **number of rooms** to "TWO" from the drop-down list.
- Change the **run creation mode** to "UPDATE_ALL" or "FROM_SCRATCH".

Using the Web Manager on the central G6 switch or CLC:

- Click the button **apply to running configuration** to save the changes to the running configuration.
- In the section **App.smartdirector** at the bottom of the window hit the button **run** to install the GUI changes.
 - » After the installation process finishes successfully, the Web GUI shows two additional buttons "Room 1" and "Room 2" when updated.

Note:

By default the Smart Director is designed to access up to 12 remote rooms or zones, depending on the selection "ONE" to "TWELVE" in the drop-down list.

- Select the **Userinterfaces** screen, then select the tab **Web**.
- In the section **Management.web.gui_element** the element "BTN_Rooms" contains all necessary values for both the buttons "Room 1" and "Room 2".

#	Parameter	Value
	name	BTN_Rooms
	type	HYPER_LINK
	page	SceneControl
	visibility	NORMAL
	auto save	<input type="checkbox"/>
	remote accessible	<input type="checkbox"/>
	sensor attribute	ROOM
	script name	
	watched element	
34	order	29
	height	25
	width	80
	top margin	5
	left margin	10
	header	
	text	Room 1,Room 2
	value	Room1Control,Room2Control
	start value	
	image	smartdirector/desk.svg,smartdirector/desk.svg
	options	frame=FRM_SceneControl, cbackground=transparent, cboxpassive=tra

Fig. 6: Web Manager - User Interface - Web - New Room Buttons

2.3 Configuring MQTT Broker and MQTT Clients

The following steps will configure the MQTT infrastructure needed to distribute the click value of a button to the respective remote Smart Director via the MQTT broker in the following order:

1. Setup of the MQTT broker
 - This example uses the local broker of the central device. If enabled this local broker is

accessible for both local (on the central device as well) and remote (on other devices) MQTT clients.

2. Setup of the central publishing Smart Director
The Smart Director running on the central device acts as MQTT client publishing its messages to the local MQTT broker.
3. Setup of the remote subscribing Smart Directors
The Smart Directors running on remote G6 switches or CLCs act as remote MQTT clients. They will subscribe to the topic concerning the remote control via the local Smart Director.

2.3.1 Using Variables

For setting up topics or IDs more easily it is possible to use specific variables. The following variables with their respective value are available:

- {SMO}: fixed text "SmartOffice"
- {MFG}: fixed manufacturer's name (i.e. "MICROSENS")
- {MAC}: MAC address of the device
(Device.factory.device_mac, e.g. "00:60:A7:09:37:4E")
- {IP4}: IPv4 address of this device
(Device.ip.v4_status.dynamic_device_ip, e.g. "10.100.89.187")
- {IP6}: IPv6 address of this device
(if enabled, Device.ip.v6_status.ip, e.g. "fe80::260:a7ff:fe09:374e/64")
- {DMN}: Domain name of the Smart Office network
(Device.smartoffice.director_config.domain_name, e.g. "domain1")
- {ART}: article number of this device
(Device.factory.article_number, e.g. "MS652119PM")
- {SER}: serial number of this device
(Device.factory.serial_number, e.g. "00345860")
- {LOC}: SNMP SysLocation
(Management.snmp.device_info.sys_location, e.g. "Office")
- {NAM}: SNMP SysName
(Management.snmp.device_info.sys_name, e.g. "MICROSENS G6 Micro Switch")

The variables can be combined e.g. in topics like "{SMO}/{MFG}_{MAC}/".

Note:

Most of the variables are device specific. E.g. using the device's IP address (i.e. "{IP4}") as topic prefix will help the broker to clearly assign subscriptions. On the other hand, the same variable for multiple devices (e.g. "{MFG}") can ensure that MQTT messages will only be utilised by specific groups of devices.

2.3.2 Setting up the Local MQTT Broker

Using the Web Manager on the central G6 switch or CLC:

- Open the Web Manager of the respective switch or CLC.

Using the Web Manager on the central G6 switch or CLC:

- Select the **MQTT** screen, then select the tab **Basic Configuration**.

Parameter	Value
enable mqtt	<input checked="" type="checkbox"/>
help topic variables	<input type="text" value="start action"/>

refresh apply to running configuration

Fig. 7: MQTT - Basic Configuration - Enable MQTT

- In the section **Protocol.mqtt** check the option **enable mqtt**. This enables all MQTT related functions of the local G6 switch or CLC.
- Click the button **apply to running configuration** to save the changes to the running configuration.
- Select the tab **Local Broker**.
- In the section **Protocol.mqtt.local_broker_config** check the option **enable local broker**.

Parameter	Value
enable local broker	<input checked="" type="checkbox"/>
permit anonymous	<input type="checkbox"/>
anonymous topic	<input type="text"/>
permit websockets	<input checked="" type="checkbox"/>
permit crypted sockets	<input checked="" type="checkbox"/>
permit encrypted ssl	<input type="checkbox"/>
permit certified ssl	<input type="checkbox"/>
certificate name	<input type="text"/>

refresh apply to running configuration

Fig. 8: MQTT - Local Broker - Enable

Note:

In this example only the publishing and subscribing MQTT clients for this remote control project should have access to the broker. Uncheck the option **permit anonymous** to prevent every non-registered MQTT client accessing the broker.

- Click the button **apply to running configuration** to save the changes to the running configuration.

Using the Web Manager on the central G6 switch or CLC:

- Scroll down to section **Protocol.mqtt.local_broker_user**. For a new element entry insert a **unique value username** in the textbox at the bottom of the user list and hit the button "+". After the confirmation dialogue the new user appears at the end of the user list, containing default values.

Add the following 3 entries:

- "Local_SD"
This is the local MQTT client that will publish the button clicks on the central Smart Director GUI as messages.
- "Remote_SD_1"
This is the remote MQTT client that will subscribe to button "Room 1" on the central Smart Director GUI.
- "Remote_SD_2"
This is the remote MQTT client that will subscribe to button "Room 2" on the central Smart Director GUI.

- In the section **Protocol.mqtt.local_broker_user** scroll down to the new user entries. Enter a **password** for each user. Hit the respective button **start action** to send the password to the device and encrypt it.

#	Parameter	Value
1	username	Local_SD
	enter password	<input type="text"/> start action
	encrypted password	<input type="text"/>
	permitted topic	RemoteSD/<Central Device>/gui/+1/SCENE
	topic access	FULL
2	username	Remote_SD_1
	enter password	<input type="text"/> start action
	encrypted password	<input type="text"/>
	permitted topic	RemoteSD/<Central Device>/gui/BTN_Room1_Scenes/1/SC
	topic access	SUBSCRIBE
3	username	Remote_SD_2
	enter password	<input type="text"/> start action
	encrypted password	<input type="text"/>
	permitted topic	RemoteSD/<Central Device>/gui/BTN_Room2_Scenes/1/SC
	topic access	SUBSCRIBE

add table entry unique value for username +

refresh apply to running configuration

Fig. 9: MQTT - Local Broker - Users

Using the Web Manager on the central G6 switch or CLC:

- Enter the following entries for **permitted topic** for the respective user:

Note:

Depending on the MQTT client type the use of wildcards for topics is allowed and helpful.

- Local_SD: "RemoteSD/<Central Device>/gui/+/1/SCENE"
The local Smart Director as publishing client should access all topic levels below "RemoteSD/gui/" containing the levels "1/SCENE".
- Remote_SD_1: "RemoteSD/<Central Device>/gui/BTN_Room1_Scenes/1/SCENE"
The subscribing client in room1 will receive only messages when the button **Room 1** is hit.
- Remote_SD_2: "RemoteSD/<Central Device>/gui/BTN_Room2_Scenes/1/SCENE"
The subscribing client in room1 will receive only messages when the button **Room 2** is hit.

Note:

The topic level entry "<Central Device>" in this example refers to a unique identifier of the central G6 switch or CLC. Insert here the device's MAC or IP address or its SysName.

Note:

The starting topic level "RemoteSD" is the topic prefix defined on tab **Basic Configuration**. Omit this level if no topic prefix is determined.

- Choose the following entries for **topic access** for the respective user:

- Local_SD: "FULL"
The local Smart Director will have full access to the determined topic structure.
- Remote_SD_1: "SUBSCRIBE"
The remote Smart Director in room 1 only will have subscribing access to the topic "RemoteSD/<Central Device>/gui/BTN_Room1_Scenes/1/SCENE".
- Remote_SD_2: "SUBSCRIBE"
The remote Smart Director in room 2 only will have subscribing access to the topic "RemoteSD/<Central Device>/gui/BTN_Room2_Scenes/1/SCENE".

- Click the button **apply to running configuration** to save the changes to the running configuration.

2.3.3 Setting up the Local MQTT Client

Using the Web Manager on the central G6 switch or CLC:

- Select the **MQTT** screen, then select the tab **Basic Configuration**.

Parameter	Value
enable publishing	<input checked="" type="checkbox"/>
topic prefix	RemoteSD
quality of service	AT_MOST_ONCE
publish all actorgroups	<input type="checkbox"/>
publish all sensorgroups	<input type="checkbox"/>
publish all actors	<input type="checkbox"/>
publish all sensors	<input type="checkbox"/>
publish gui elements	<input checked="" type="checkbox"/>

refresh apply to running configuration

Fig. 10: MQTT - Basic Configuration - Enable MQTT Publishing

- In the section **Protocol.mqtt.publisher_config** make the following changes and entries:
 - Check the option **enable publishing**.
This enables all MQTT publishing functions like generating valid MQTT messages.
Note:
This forces the MQTT client to publish every mouse click's respective GUI value to the broker. It is not essential to map every click to a specific topic for publishing.
 - Change the **topic prefix** to a descriptive name (e.g. "RemoteSD").
This will prepend this prefix to every outgoing topic in the form "prefix/<rest of topic levels>".
Note:
The permitted topics for broker users must include this prefix.
Note:
It is possible to use topic variables (e.g. "{SMO}", "{MFG}", "{DMN}" etc.). Keep in mind that these variables are bound to the central device. On remote devices these topic variables could have other values so that subscribing MQTT clients using these variables may not access the correct topic managed by the broker on the central device.
 - Check the option **publish gui elements**.
This will transmit the GUI element activity (i.e. click on one of the buttons "Room 1" or "Room 2").
- Click the button **apply to running configuration** to save the changes to the running configuration.

Using the Web Manager on the central G6 switch or CLC:

- Select the tab **Broker Access**.

Parameter	Value
broker address	<input type="text"/>
client id	<input style="background-color: #f0f0f0;" type="text" value="{MFG}_{MAC}"/>
security mode	<input style="background-color: #f0f0f0;" type="text" value="NONE"/>
username	<input style="background-color: #f0f0f0;" type="text" value="Local_SD"/>
password	<input type="password"/>
keep alive timer	<input style="background-color: #f0f0f0;" type="text" value="180"/>
trace mode	<input type="checkbox"/>

Fig. 11: MQTT - Broker Access

- In the section **Protocol.mqtt.broker_access** make the following changes and entries:
 - Enter the **username** and **password** for secure broker access. Use the credentials for the local user "Local_SD" how they were defined as broker user on tab **Local Broker**.
 - Enabling the **trace mode** for debugging reasons is optional. If enabled the MQTT messages are visible via CLI.

Note:

Since the central Smart Director (as MQTT client) and the MQTT broker are on the same G6 switch or CLC leave the field **broker address** empty. This assigns the local broker as MQTT broker.

- Click the button **apply to running configuration** to save the changes to the running configuration.

2.3.4 Setting up the Remote MQTT Clients

The central G6 switch or CLC is ready for publishing MQTT messages:

- The local Smart Director acts as publishing MQTT client sending its messages to the local MQTT broker.
- The local MQTT broker forwards these messages to MQTT clients which have subscribed to the respective topics.

The following steps describe the setup for the remote MQTT clients to subscribe to specific topics published by the broker.

Using the Web Manager on the remote G6 switch or CLC:

- Open the Web Manager of the respective switch or CLC.

Using the Web Manager on the remote G6 switch or CLC:

- Select the **MQTT** screen, then select the tab **Basic Configuration**.

Parameter	Value
enable mqtt	<input checked="" type="checkbox"/>
help topic variables	start action

refresh apply to running configuration

Fig. 12: MQTT - Basic Configuration - Enable MQTT

- In the section **Protocol.mqtt** check the option **enable mqtt**. This enables all MQTT related functions of the G6 switch or CLC.
- Click the button **apply to running configuration** to save the changes to the running configuration.
- Select the tab **Broker Access**.

Parameter	Value
broker address	192.168.1.90
client id	<device name>
security mode	NONE
username	Remote_SD_1
password	
keep alive timer	180
trace mode	<input type="checkbox"/>

refresh apply to running configuration

Fig. 13: MQTT - Broker Access

- In the section **Protocol.mqtt.broker_access** make the following changes and entries:
 - Enter the **broker address** of the MQTT broker. This is the IPv4 address of the central G6 switch or CLC. It is not necessary to specify a port. MQTT by default communicates over port 1883.
 - Enter the **client id** of this MQTT client.
 - Enter the **username** and **password** for secure broker access. Use the credentials for the remote users as they were defined as broker user on tab **Local Broker**. on the central device:
 - Remote device in room 1: "Remote_SD_1" and respective password
 - Remote device in room 2: "Remote_SD_2" and respective password
 - Enabling the **trace mode** for debugging reasons is optional. If enabled the MQTT messages are visible via the CLI of the respective device.

Using the Web Manager on the remote G6 switch or CLC:

- Click the button **apply to running configuration** to save the changes to the running configuration.
- Select the tab **Topics**. For a new element entry insert a **unique value for name** in the textbox at the bottom of the topic list (e.g. "Global_SCENE") and hit the button "+". After the confirmation dialogue the new topic appears at the end of the topic list, containing default values.
- In the section **Protocol.mqtt.topic_map** scroll down to the new topic entry.

Protocol.mqtt.topic_map (1 entry)		
#	Parameter	Value
1	name	Global_SCENE
	mode	SUBSCRIBE
	topic	RemoteSD/<Central Device>/gui/BTN_Room1_Scenes/1/SCE
	local name	BTN_Scenes:1:SCENE
	quality of service	AT_LEAST_ONCE

add table entry unique value for name +

refresh apply to running configuration

Fig. 14: MQTT - Topics

- Make the following changes and entries:
 - Change the **mode** for this topic to "SUBSCRIBE".
 - Enter the **topic** for the subscription. This entry must match the determined topic on the broker for this specific client.

Note:
This entry must contain the topic prefix as it is defined in the publishing MQTT client.

 - Enter the **local name** for this topic entry ("BTN_Scenes:1:SCENE"). This will send the received value of the subscribed topic to the same GUI element which is controlled in the Web GUI.

Note:
This entry is identical on both remote MQTT clients.
- Click the button **apply to running configuration** to save the changes to the running configuration.

Disclaimer

All information in this document is provided 'as is' and subject to change without notice. MICROSENS GmbH & Co. KG disclaims any liability for the correctness, completeness or quality of the information provided, fitness for a particular purpose or consecutive damage.

Any product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

©2021 MICROSENS GmbH & Co. KG, Kueferstr. 16, 59067 Hamm, Germany.

All rights reserved. This document in whole or in part may not be duplicated, reproduced, stored or retransmitted without prior written permission of MICROSENS GmbH & Co. KG.

Document ID: AN-EN-21003