

# RUT900 MQTT (legacy WebUI)

[Main Page](#) > [RUT Routers](#) > [RUT900](#) > [RUT900 Manual](#) > [RUT900 Legacy WebUI](#) > [RUT900 Services section \(legacy\)](#) > **RUT900 MQTT (legacy WebUI)**

The information in this page is updated in accordance with firmware version **[RUT9XX\\_R\\_00.06.09.5](#)**.

**Note:** this user manual page is for RUT900's old WebUI style available in earlier FW versions. [Click here](#) for information based on the latest FW version.



## Contents

- [1 Summary](#)
- [2 MQTT Broker](#)
  - [2.1 Security](#)
  - [2.2 Bridge](#)
    - [2.2.1 Topics](#)
  - [2.3 Miscellaneous](#)
- [3 MQTT Publisher](#)

## Summary

**MQTT (MQ Telemetry Transport or Message Queue Telemetry Transport)** is an ISO standard (ISO/IEC PRF 20922) publish-subscribe-based "lightweight" messaging protocol for use on top of the TCP/IP protocol. It is designed to send short messages from one client (**publisher**) to another (**subscriber**) through **brokers**, which are responsible for message delivery to the end point.

RUT9XX routers support this functionality via an open source Mosquitto broker. The messages are sent this way: a client (**subscriber**) subscribes to a topic(s); a publisher posts a message to that specific topic(s). The **broker** then checks who is subscribed to that particular topic(s) and transmits data from the publisher to the subscriber.

The following chapter is a summary of the MQTT function in RUT9XX routers. For in-depth MQTT configuration examples, refer to this page: [Monitoring via MQTT](#).

## MQTT Broker

The **Broker** will "listen" for connections on the specified Local port. In order to accept connections from WAN, you also need to check Enable Remote Access.



Field name	Value	Description
Enable	yes   no; default: <b>no</b>	Toggles MQTT Broker ON or OFF.

Local Port	integer [0..65535]; default: <b>1883</b>	Specifies the local port that the MQTT broker will listen to.
Enable Remote Access	yes   no; default: <b>no</b>	If enabled, MQTT Broker will be reachable by remote user (from WAN).

## Security

---

The MQTT **Security** tab is used to establish MQTT connection security via TLS/SSL.



Field name	Value	Description
Use TLS/SSL	yes   no; default: <b>no</b>	Toggles the use of TLS/SSL certificates ON or OFF.
CA File	.ca file; default: <b>none</b>	<b>Certificate authority</b> is an entity that issues digital certificates. A digital certificate certifies the ownership of a public key by the named subject of the certificate.
CERT File	.crt file; default: <b>none</b>	Certificate file is a type of digital certificate that is used by client systems to make authenticated requests to a remote server. Client certificates play a key role in many mutual authentication designs, providing strong assurances of a requester's identity.
Key File	.key file; default: <b>none</b>	Private key for client to establish connection.
TLS version	tlsv1.1   tlsv1.2   Support all; default: <b>Support all</b>	Authenticates a client to a server and establishes precisely who they are.

## Bridge

---

The MQTT Broker also supports a functionality called **Bridge**. An MQTT Bridge is used for the communication between two MQTT Brokers. The window of Bridge parameters is presented below. Some of these are mandatory as they are needed to create a connection: Connection Name, Remote Address and Remote Port. For more information on **MQTT Bridge** parameters you can read the official [mosquitto.conf](https://mosquitto.org/man/mosquitto-conf.5.html) manual page.



Field name	Value	Description
Use TLS/SSL	yes   no; default: <b>no</b>	Toggles MQTT Bridge ON or OFF.
Connection Name	string; default: <b>none</b>	Name of the Bridge connection. Although this is used for easier management purposes, this field is mandatory.
Remote Address	ip; default: <b>none</b>	Remote Broker's address.
Remote Port	integer [0..65535]; default: <b>1883</b>	Specifies which port the remote broker uses to listen for connections.

Use Remote TLS/SSL	yes   no; default: <b>no</b>	Enables the use of TLS/SSL certificates of the remote broker. If this is checked, you will be prompted to upload TLS/SSL certificates. More information can be found in the <a href="#">Security</a> section of this chapter.
Use Remote Bridge Login	yes   no; default: <b>no</b>	Enables the use of Remote login data. If this is checked, you will be prompted to enter a remote client ID, username and password.
Topic	string; default: <b>none</b>	Specifies the names of the Topics that your Broker will subscribe to.
Try Private	yes   no; default: <b>no</b>	Check if the remote Broker is another instance of a daemon.
Clean Session	yes   no; default: <b>no</b>	Check to discard session state after connecting or disconnecting.

## Topics

---

The **Topic** section is used to add and configure topics for MQTT communication.

✖

Field	Value	Description
Topic	string; default: <b>none</b>	The name of this topic.
Direction	OUT   IN   BOTH; default: <b>OUT</b>	Specifies the direction in which messages for this topic can be shared in.
QoS level	integer; default: <b>0</b>	The publish/subscribe QoS level for this topic..
Delete	-(interactive button)	Deletes the topic next to it.

## Miscellaneous

---

The last section of MQTT Broker parameters is called **Miscellaneous**. It contains parameters that are related to neither Security nor Bridge.

✖

field name	value	description
ACL File	.ACL file; default: <b>none</b>	The contents of this file are used to control client access to topics of the broker.
Password File	password file; default: <b>none</b>	The Password file stores user names and corresponding passwords, used for authentication. The file can be produced by mosquitto_passwd on the ssh command line, see <a href="https://mosquitto.org/man/mosquitto_passwd-1.html">https://mosquitto.org/man/mosquitto_passwd-1.html</a>
Persistence	yes   no; default: <b>no</b>	If enabled, connection, subscription and message data will be written to the disk. Otherwise, the data is stored in the router's memory only.
Allow Anonymous	yes   no; default: <b>yes</b>	If enabled, the Broker allows anonymous access.

# MQTT Publisher

An **MQTT Publisher** is a client that sends messages to the Broker, who then forwards these messages to the Subscriber.



field name	value	description
Enable	yes   no; default: <b>no</b>	Toggles the MQTT Publisher ON or OFF.
Hostname	host   ip; default: <b>none</b>	Broker's IP address or hostname.
Port	integer [0..65535]; default: <b>1883</b>	Specifies the port used for connecting to the Broker.
Username	string; default: <b>none</b>	User name used for authentication when connecting to the Broker.
Password	string; default: <b>none</b>	Password used for authentication when connecting to the Broker.
TLS	yes   no; default: <b>no</b>	Toggles the TLS authentication between ON or OFF.
Allow insecure connection	yes   no; default: <b>no</b>	If enabled, router will be allowed to not verify server's authenticity.
CA file	.ca file; default: <b>none</b>	<b>Certificate authority</b> is an entity that issues digital certificates. A digital certificate certifies the ownership of a public key by the named subject of the certificate.
Certificate file	.crt file; default: <b>none</b>	Certificate file is a type of digital certificate that is used by client systems to make authenticated requests to a remote server. Client certificates play a key role in many mutual authentication designs, providing strong assurances of a requester's identity.
Key file	.key file; default: <b>none</b>	Private key for client to establish connection.