

Chariot MQTT Client

Prerequisites

- [Install](#) the Chariot MQTT Server
 - The Chariot MQTT Client does not require a Chariot license or that you run Chariot under its trial timer.

Summary

The Chariot MQTT Client is a built-in MQTT application that allows you to connect to any number of MQTT Servers simultaneously, subscribe on any topic (s) across the MQTT servers, publish MQTT messages and view/interact with the MQTT topic tree as messages are published. The application allows for full control of MQTT QoS levels and the retain flag.

The Chariot MQTT Client consists of three tabbed views:

- The Connections tab where you can create and manage your MQTT server connections.
- The Publish/Subscribe tab where you can subscribe on MQTT topics and produce MQTT messages on any topic with a custom payload. The Device Simulator also provides a quick way to get test messages published. The Messages view will decode plaintext, SparkplugB, Hex and Base64.
- The Topic Tree Viewer tab where you can view all messages across all topics simultaneously across multiple MQTT servers with live decoding of the messages.

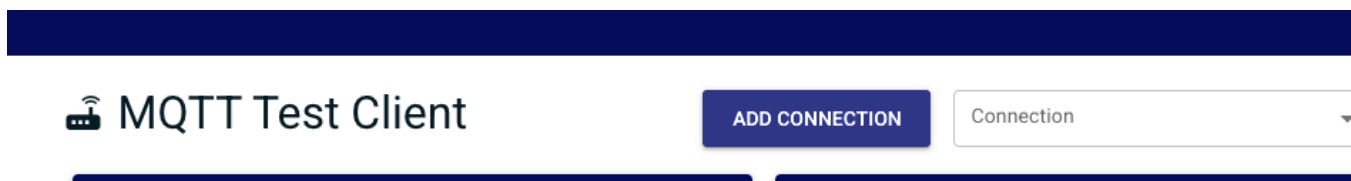
Launch the Chariot MQTT Client

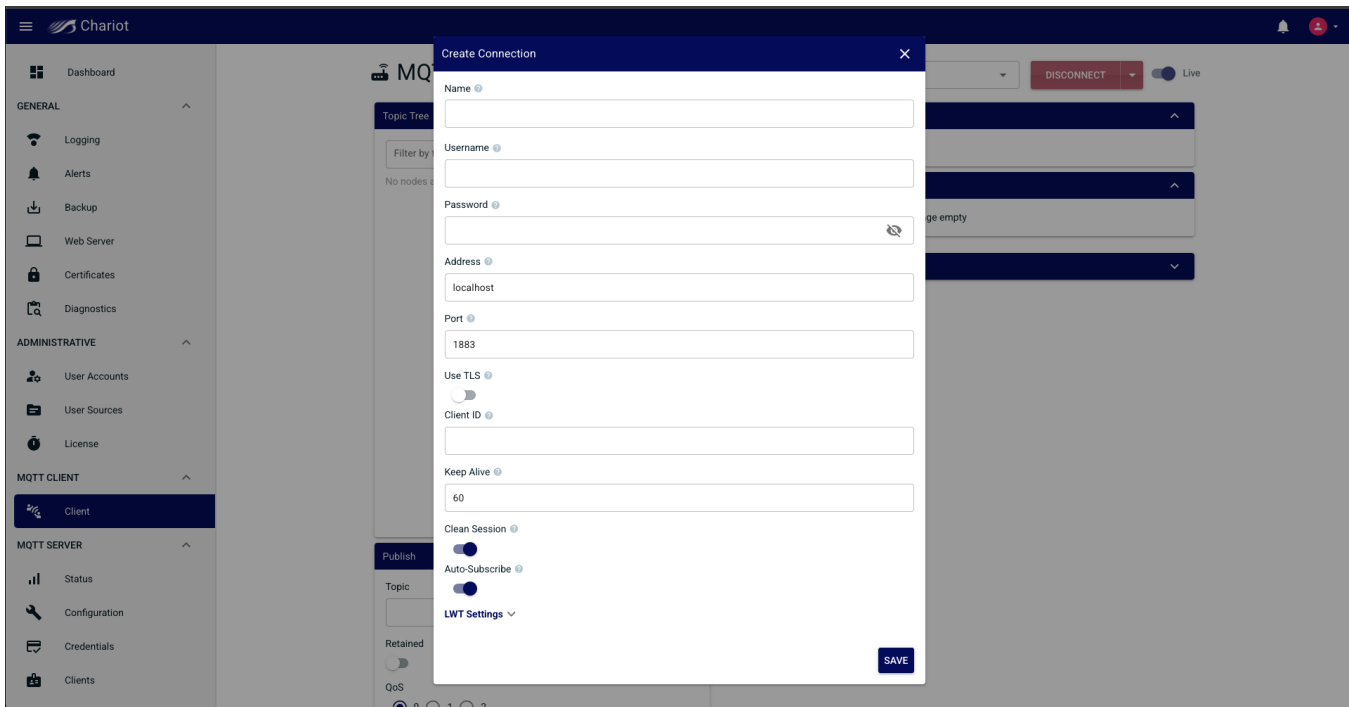
To launch the Chariot MQTT Client, first log into the Chariot MQTT server web front-end. Then click on the MQTT Client View under **Main Nav MQTT Test Client** on the Chariot left-hand navigation pane.



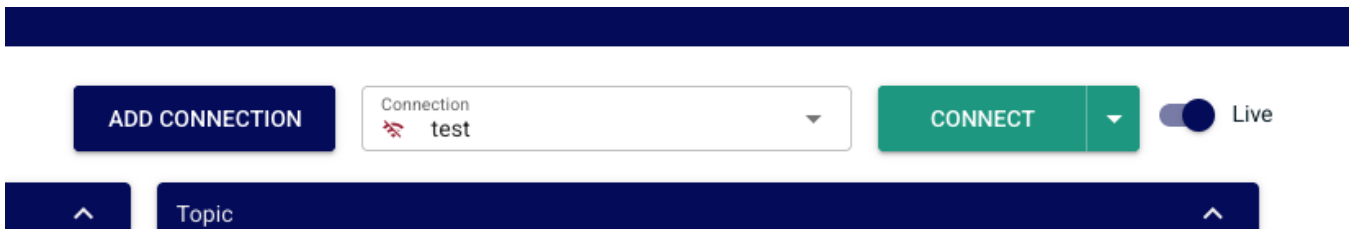
Add Server Connection

To get started, navigate to the Connections tab and add a server connection by clicking on the Add Connection button. Fill out the Connection configuration (MQTT username, password, address/url, port, client id, etc) and then click the Add button. This can be any MQTT server (e.g., test.mosquitto.org). If you do want to connect to the local Chariot MQTT Server using the Chariot MQTT Client, please be sure to turn on the trial timer so the server will accept incoming connections. See our [documentation](#) for default MQTT account credentials.





Next, click the connect icon to connect to the newly added MQTT server. Alternatively you can use the dropdown in the header to select from the available connections and clicking on the connect/disconnect buttons. Your current MQTT connection and its state will be displayed in the upper right of the window.



Publish and Subscribe

Next, move to the Publish/Subscribe tab to setup your MQTT subscriptions and publish messages if desired. For example, subscribe on '#' in the Subscribe view - set the topic to '#' and click the Subscribe button.

- Dashboard
- GENERAL
 - Logging
 - Alerts
 - Backup
 - Web Server
 - Certificates
 - Diagnostics
- ADMINISTRATIVE
 - User Accounts
 - User Sources
 - License
- MQTT CLIENT
 - Client**
- MQTT SERVER
 - Status
 - Configuration
 - Credentials
 - Clients

MQTT Test Client

ADD CONNECTION

Connection test

DISCONNECT

Live

Topic Tree

Filter by topic CLEAR MESSAGES

- test (2) (localhost:1883)
 - spBV1.0
 - STATE
 - testtest (1)
 - test (1)

Topic

CLEAR TOPIC

SPBV1.0 / STATE / TESTTEST

Most Recent Message

Encoding: Plaintext (UTF-8) RETAINED 04/03/2026, 08:00:44.479 QoS: 0

```
{  
  "online": true,  
  "timestamp": 1775227594582  
}
```

History

Publish

Topic: test

Retained:

QoS: 0 1 2

Topic Tree Topic

Filter by

test (2) (0)
 spBV
 ST
test (1)

Edit Subscriptions

Topic

QoS
 0 1 2

SUBSCRIBE

Subscriptions

Topic	QoS
#	0

Records per page: 5 1-1 of 1

Publish

Topic

test

Now publish a message using the Publish view and watch it show up decoded in the Messages view.

Publish



Topic

test

Retained



QoS

0 1 2

Message Type

Plaintext Message Device Simulator

Message

test

PUBLISH

Most Recent Message ^

Encoding
Plaintext (UTF-8) ▼

RETAINED

04/03/2026, 08:00:44.479
QoS: 0



```
{  
  "online": true,  
  "timestamp": 1775227594582  
}
```

History ^

Encoding
Plaintext (UTF-8) ▼

DOWNLOAD MESSAGES

Timestamp	Topic	Payload	QoS
04/03/2026, 08:00:44.479	spBv1.0/STATE/testtest	{"online": true, "ti...	0

Records per page: 5 ▼ 1-1 of 1

View the Topic Tree

Lastly, view the complete, live topic tree with all topics and messages visible across all MQTT server connections. Simply navigate the Topic Tree view to find the messages you want by topic. Each topic will show how many messages the Chariot MQTT Client has cached locally. When you select the leaf node / last topic token, the Messages view will automatically be populated with all messages available on that topic. The subscriptions you make in the Publish/Subscribe tab apply in the Topic Tree viewer.

History



Encoding
Plaintext (UTF-8) ▼

DOWNLOAD MESSAGES

Timestamp	Topic	Payload	QoS
04/03/2026, 08:03:08.476	test	test test test test	0
04/03/2026, 08:00:49.276	test	test	0

Records per page: 5 ▼ 1-2 of 2

Compare Messages



Current Message:

test test test test



Compared Message:

test



QoS

0 1 2