# **Chariot MQTT Server Configuration**

This document describes the configuration options in the Chariot® MQTT Server UI.

The UI can be accessed at the following URL:

http://<server-url>:8080

When installed on Linux, Windows or deployed via Azure Marketplace, the default User login credentials are:

```
username: admin
password: password
```

When deployed via AWS Marketplace, the default User login credentials are:

```
username: admin
password: EC2 instance ID for example: i-0049ac1e13e558b70
```

The Chariot MQTT Server Web UI provides multiple configuration pages on the left navigation panel.

- Accounts
- MQTT Credentials
- MQTT Server
- License
- System

### Accounts

The Accounts page allows for the creating, updating and deleting of Users, Roles and Sources.

Users

The Users page allows for the creating, updating, and deleting of Chariot Users. A Chariot User may log into the Chariot Web UI and perform different actions based the the Roles assigned to that User. The main Chariot Users page shows a table of existing Chariot Users. Users can be added by clicking the "Add User" button, or they can be edited by clicking on the individual User in the table.

🔴 🔴 🔹 🖻 🎢 Chariot						
$\leftarrow \rightarrow \mathbf{G}$ (	🔿 🗟 😋 192.168.1.81:8080/#/accounts			☆	♡ ⊻ © □	മ ≓
Chariot admin ▽	Accounts					
STATUS						
📲 Dashboard	Home > Accounts					
al Logging	Users Roles Sources					
🖇 Sparkplug	Add User					
🔪 мотт	USERNAME	EMAIL	FIRST NAME	LAST NAME		
ậ Alerts	admin					
බ් Diagnostics						
			( ) $( )$			
器 Accounts						
MQTT Credentials						
MQTT Server						
ථ License						
□ System						
←						

# Add User

Clicking the "Add User" button will display a form for creating a new Chariot User.

🗕 🌢 💼 🦪 Charic	ot								
$\leftarrow \rightarrow G$	0 🗟 👓	192.168.1.81:8080/#/accounts		ជ		⊚ ₹	۲	ර =	•
Chariot admin ⊽	00	Add User							
STATUS									
🗄 Dashboard	Home >	Username	bsmith						
๗ Logging	User	Password		0					
🖇 Sparkplug	Add	Email	bsmith@example.com						
∖ mqtt	USERI	First Name	Bob		TNAME				
û Alerts	admin	Last Name	Smith						
Diagnostics									
CONFIGURATION		Roles	admin ×	*					
器 Accounts									
MQTT Credentials			Cancel	Add					
MQTT Server									
🔿 License									
🗆 System									
←									

The form contains the following fields:

- Username
- A unique username for the Chariot User (required).
   Password
- A secure password for the Chariot User that will be used to log into the Chariot Web UI (required).
- Email
- An email address associated with the Chariot User (optional).
- First Name • A first name associated with the Chariot User (optional).
- Last Name
- A last name associated with the Chariot User (*optional*).
  Roles
  - A list of Chariot Roles associated with the Chariot User.

A By default, a User with the Username admin and Password password will be created

The "Add" button in the bottom right of the form can be clicked to create the User.

### Edit User

Clicking on an individual Chariot User in the table will display a form for editing the Chariot User.

Chariot	× +						
$\leftarrow \rightarrow \mathbf{G}$	웥 ⊶ 192.168.1.81:8080/#/users/bsmith		☆	⊗ ₹	٢	C] 2	ב נ
Chariot admin ⊽	First Name	Bob					
TATUS	Last Name	Smith					
🗄 Dashboard	Email	bsmith@example.com					
d Constants	Company						
<ul> <li>✓ Sparkplug</li> <li>MQTT</li> </ul>	Street						
Ω Alerts	City						
Jiagnostics	State or Province						
	Country						
MQTT Credentials	Zip Code						
MQTT Server	Phone Number						
🕑 License	Roles	admin ×				•	
🗆 System							
←					Up	date	

The update form contains additional fields that can be used to provide more information about the Chariot User. Additionally, a dropdown list to the right of the Chariot User's username can be clicked to provide options for deleting the user, or changing the Chariot User's password.

🔴 🌢 🌒 🖻 🥂 Chariot	× +			~
$\leftarrow \rightarrow \mathbf{G}$	🗘 👌 ⊶ 192.168.1.81:8080/#/users/bsmith		☆	ଓ ଏ © ୍ ମ ≓
Chariot admin	User :			
STATUS	Change Passwo	rd		
📲 Dashboard	🗎 Delete User			
ad Logging	First Name	Bob		
🖗 Sparkplug	Last Name	Smith		
∖ MQTT		Sman		
û Alerts	Email	bsmith@example.com		
l Diagnostics	Company			
CONFIGURATION	Street			
<ul> <li>Accounts</li> <li>MQTT Credentials</li> </ul>	City			
S MQTT Server	State or Province			
🔿 License	Country			
□ System	Zip Code			
← 192.168.1.81:8080/#	Phone Number			

The "Update" button in the bottom right can be clicked to update the Chariot User once all changes have been made.

Roles

The Roles page allows for the creating, updating, and deleting of Chariot Roles. A Chariot Role is a collection of permissions that authorized viewing, editing, and/or controlling the Chariot MQTT Server. The main Chariot Roles page shows a table of existing Chariot Roles. Roles can be added by clicking the "Add Role" button, or they can be edited by clicking on the individual Role in the table.

🗕 🕘 💼 🥠 Chariot					~
$\leftarrow \rightarrow \mathbf{G}$	0 👌 🕶 192.168.1.81:8080/#/accounts?tab=roles		☆	⊚ ±	 பீ ≡ீ
Chariot admin  <	Accounts				
STATUS 旧 Dashboard	Home > Accounts				
al Logging	Users Roles Sources				
🖇 Sparkplug	Add Role				
√ ΜΩΤΤ	NAME	PERMISSIONS			
û Alerts	admin	[ ***** ]			
Diagnostics					
Accounts     MQTT Credentials					
MQTT Server					
🔿 License					
System					
←					

### Add Role

Clicking the "Add Role" button will display a form for creating a new Chariot Role.

🔍 🔍 💿 💼 🥖 Chari	ot × +			~
$\leftarrow \rightarrow G$	🔿 👌 🕶 192.168.1.81:8080/#/accounts?	tab=roles	☆	ତ ⊻ ଓ ୍ £ =
Chariot admin ⊽	Add Role			
STATUS				
🗄 Dashboard	Home > Name	read-only		
႕] Logging	User: Permissions	permission:read × license:read × server:read ×	~	
🖗 Sparkplug	Add	sparkplug:read * logging:read * user:read *		
∿ mqtt	NAME	accountread *		
ậ Alerts	admin			
Jiagnostics			Cancel Add	
CONFIGURATION				
ಜಿ Accounts				
MQTT Credentials				
🛢 MQTT Server				
🔄 License				
□ System				
÷				

The form contains the following fields:

- Name
- A unique name for the Chariot Role.
- Permissions
  - A list of permissions for the Chariot Role.
  - Permissions have the form <domain>:<action>
    - The <domain> represents a service within Chariot that can be interacted with through the Chariot UI.
      - Some examples are:
        - user
          - $^{\circ}~$  The User Service for managing Chariot UI users
          - See the Chariot UI under the "Users" tab
          - role
            - The Role Service for managing Chariot UI roles
            - See the Chariot UI under the "Roles" tab
          - mqttuser
            - The MQTT User Service for managing MQTT Credentials
            - See the Chariot UI under the "MQTT Credentials" tab
          - system
            - The System Service for configuring Server Name, HTTP, Certificates, and Backup/Restore
               See the Chariot UI under the "System" tab
          - server
            - The MQTT Server Service for configuring and controlling the MQTT server
               See the Chariot UI under the "MQTT Server" tab
            - ° See
          - license
            - The Licensing Service for licensing the Chariot MQTT Server software
            - See the Chariot UI under the "License" tab
      - The <action> represents the scope of this permission for the associated domain.
      - The available actions are:
        - read
          - The permission to read/view resources, configurations, and/or data
        - create
          - The permission to create new resources and/or configurations
        - update
          - The permission to update resources and/or configurations
        - delete
          - The permission to delete resources and/or configurations
        - action
          - The permission to perform any actions available from a service within Chariot (such as starting/stopping the MQTT server or activating a license)

The "Add" button in the bottom right of the form can be clicked to create the Role.

### **Edit Role**

Clicking on an individual Chariot Role in the table will display a form for editing the Chariot Role.

🗕 🌢 💼 📝 Chari	bt × +					
$\leftarrow \  \   \rightarrow \  \   G$	🔘 🗟 🕶 192.168.1.81:8080/#/roles/read-only		☆	⊗⊁e	) 📫 එ	) ≡
Chariot admin ▽	ကြာ Role :					
STATUS						
📲 Dashboard	Home > Accounts > read-only					
al Logging	Permissions	permission:read × license:read × set	erver:read ×) role:read ×) system:create ×	3	*	
🎸 Sparkplug		mqttuser:read * sparkplug:read *	logging:read * user:read * account:read	×		
∖ MQTT					Update	
û Alerts				_		
Diagnostics						
ONFIGURATION						
恩 Accounts						
MQTT Credentials						
MQTT Server						
් License						
□ System						
←						

The "Update" button in the bottom right can be clicked to update the Chariot Role once all changes have been made.

### Sources

The Chariot Security Service can be configured to add an LDAP Realm to use when authenticating and authorizing access via the Chariot UI.

This feature is available in Chariot v2.4.2 and newer

Review the LDAP and Microsoft Active Directory for Chariot UI/REST Users for configuration details

• • • 🖻 🥖 Chariot	× +				~
$\leftarrow \rightarrow G$	🔿 👌 ⊶ 192.168.1.81:8080/#/a	ccounts?tab=sources		☆	ଓ ⊻ 🖲 🛱 📬
Chariot admin	Accounts				
STATUS	Llama > Assausta				
📲 Dashboard	Home > Accounts				
പി Logging	Users Roles Source	s			
🖗 Sparkplug	Add Source				
ν μάιτ	NAME	TYPE	ENABLED		
û Alerts					
l Diagnostics			No sources found		
CONFIGURATION			no sources round		
器 Accounts					
MQTT Credentials					
MQTT Server					
් License					
System					

# **MQTT** Credentials

The MQTT Credentials page contains two tabs: MQTT Credentials and LDAP

MQTT Credentials are the credentials that clients will use to authenticate when establishing an MQTT connection to the Chariot MQTT Server.

MQTT Credentials also contain the access control lists (ACLs) that control which MQTT topics a client can publish and/or subscribe on. Credentials can be added by clicking the "Add Credential" button, or they can be edited by clicking on the individual Credential in the table.

From Chariot release 2.5.0, to avoid manual configuration, the required ACLs can be generated from the STATUS > Sparkplug > Tools menu and will automatically be added to the MQTT Credentials

Chariot	×	+	~	
$\leftarrow \rightarrow \mathbf{G}$ O	🔏 O <del>.,</del> 192.168.	1.81:8080/#/mqttcredentials	않 오구 🖲 🗇	<b>=</b>
Chariot admin  ▽		2TT Credentials ©		
STATUS		redentials		
📲 Dashboard		i eneritai a		
al Logging	Credentials	LDAP		
∲ Sparkplug	Add Credent	ial		
∖ MQTT	USERNAME	PUBLISH TOPICS	SUBSCRIBE TOPICS	
Alerts	primaryhostl	STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/#, spBv1.0/G2/+/ E1/#, spBv1.0/G3/+/E1/#, spBv1.0/STATE/MyTestHost	STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/#, spBv1.0/G2/+/ E1/#, spBv1.0/G3/+/E1/#, spBv1.0/STATE/MyTestHost	
CONFIGURATION	test	spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/#	STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/#, spBv1.0/ STATE/MyTestHost	
ಜ Accounts	admin	#	#	
MQTT Credentials				
MQTT Server			,	
🔄 License				
□ System				
÷				

# Add Credential

Clicking the "Add Credential" button will display a form for creating a new MQTT Credential.

← → C O d Chariot admin ✓ TATUS IB Dashboard	Add Credential       Orne ≥     Username	testcred	☆ ×	∞ 7	9 🗆 L
Charlot admin  Charlot Admin TATUS CE Dashboard	Add Credential	testcred	×		
TATUS	ome > Username 💿	testcred			
Il Logging					
	Cred Password @		۵		
5 Sparkplug	Add Publish Topics 💿	test/pub/#	Add		
∖ мотт	USERI Subscribe Topics 📀	test/sub/#	Add		
Alerts	primar		:1/# TAT	;, spBv1.0/G1/+/E4/#, spl E/MyTestHost	3v1.0/G2/+/
NFIGURATION	test	STATE/My	Cancel Add TestHost	;, spBv1.0/G1/+/E4/#, spl	3v1.0/
8 Accounts	admin #	#			
MQTT Credentials					
MQTT Server					
ා License					
System					
←					

The form contains the following fields:

- Username
- A unique username that an MQTT client will use when connecting.
- Password
- A secure password that an MQTT client will use when connecting.
   Publish Topics
- A list of MQTT topic filters that the client is allowed to publish on (wildcards may be used).
- Subscribe Topics
  - A list of MQTT topic filters that the client is allowed to subscribe on (wildcards may be used).

By default, two default MQTT Credentials are created:

Username admin Password changeme

Username anonymous Password changeme

The "Add" button in the bottom right of the form can be clicked to create the Credential.

Topic Examples:

- #
  - Allows publish or subscribe on all topics
- STATE/123
- Allows publish or subscribe on the specific topic "STATE/123"
   spBv1.0/#
  - Allows publish or subscribe on all topics that start with "spBv1.0/", such as "spBv1.0/g1/DDATA/e1/d1"

Examples of MQTT Credentials

I Chariot			
← → C O A or 1	2.168.1.81:8080/#/mqttcredentials	☆	=
Chariot admin	MQTT Credentials 💿		
STATUS			
Dashboard	QTT Credentials		
al Logging Creder	tials LDAP		
	edential		
\ MQTT USERN	ME PUBLISH TOPICS	SUBSCRIBE TOPICS	
	spBv1.0/g1/NBIRTH/e1, spBv1.0/g1/DBIRTH/e1/#, spBv1.0/g1/NDATA/e1, spBv1.0/g1/ Edgel DDATA/e1/#, spBv1.0/g2/NBIRTH/e2, spBv1.0/g2/DBIRTH/e2/#, spBv1.0/g3/NBIRTI #, spBv1.0/g3/DBIRTH/#, spBv1.0/g3/NDATA/#, spBv1.0/g3/DDATA/#	<ul> <li>STATE/123, spBv1.0/g1/NCMD/e1, spBv1.0/g1/DCMD/e1/#,</li> <li>spBv1.0/g2/NCMD/e2, spBv1.0/g2/DCMD/e2/#, spBv1.0/ g3/NCMD/#, spBv1.0/g3/DCMD/#</li> </ul>	
CONFIGURATION	test/pub/#	test/sub/#	
Accounts     MQTT Credentials	ost1 STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/#, spBv1.0/G2/+/E1/#, spBv1.0/G3/+/E1/#, spBv1.0/STATE/MyTestHost	STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/ #, spBv1.0/G2/+/E1/#, spBv1.0/G3/+/E1/#, spBv1.0/STATE/ MyTestHost	
S MQTT Server test	sp8v1.0/G1/+/E1/#, sp8v1.0/G1/+/E4/#	STATE/MyTestHost, spBv1.0/G1/+/E1/#, spBv1.0/G1/+/E4/ #, spBv1.0/STATE/MyTestHost	
ී License admin	#	#	
□ System ←	( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( )		

### **Edit Credential**

Clicking on an individual MQTT Credential in the table will display a form for editing the Credential.

🗧 🔵 🔹 📝 Charic				
$\leftarrow \rightarrow \mathbf{C}$	○ 웝 ⊶ 192.168.1.81:8080/#/mqttcredentia	als/test	☆	ල ⊻ ල 🕮 එ ≓
Chariot admin ▽	MQTT Credential	:		
STATUS				
📲 Dashboard	Home > MQTT Credentials > test			
.๗ Logging	Publish Topics	spBv1.0/G1/+/E1/# × spBv1.0/G1/+/E4/# × Enter To	opic	
🖗 Sparkplug				
∖ MQTT	Subscribe Topics	STATE/MyTestHost * spBv1.0/Ci/+/Ei/# * spBv1.0/ Enter Topic	/G1/+/E4/# * spBv1.0/STATE/MyTestHo	st *
↓ Alerts				
Diagnostics				Update
CONFIGURATION				
器 Accounts				
MQTT Credentials				
MQTT Server				
ව License				
🗆 System				

The "Update" button in the bottom right can be clicked to update the Credential once all changes have been made.

### LDAP

Chariot can be configured to use an LDAP server for MQTT client authentication and authorization instead of Chariot's MQTT Credentials.

This feature is available in Chariot v2.4.2 and newer Review the LDAP for MQTT Clients for configuration details

🗧 🌢 💿 💼 🎻 Chario	1 × +			~
$\leftarrow \rightarrow C$	O 👌 ⊶ 192.168.1.81:8080/#/mqttcredent	tials?tab=ldap	☆	ତ ± @ ୍ ମ ≓
Chariot admin ⊽	MQTT Credentials	• ®		
STATUS				
📲 Dashboard	Home > MQTT Credentials			
al Logging	Credentials LDAP			
🖗 Sparkplug	(X) Not connected			
∖ MQTT				
û Alerts	Enabled 💿			
Diagnostics	Host 🕜	localhost		
CONFIGURATION	Port	389		
器 Accounts				
MQTT Credentials	Use ILS 🔮			
MQTT Server	System Username 💿			
් License	System Password 💿			
□ System	Base DN 🥥			
+	Username Attribute Name 💿			

# **MQTT Server**

The MQTT Server page contains two tabs: Configuration and Bridging.

# Configuration

The configuration tab is a simple form used to configure the MQTT Server.

🗕 🕒 💼 🎻 Chariot	× +			~
$\leftarrow \rightarrow \mathbf{G}$	🗘 👌 🕶 192.168.1.81:8080/#/mqttserver		☆ ♡±	ി മ ≓
Chariot admin ▽	MQTT Server 💿			
STATUS []: Dashboard	Home > MQTT Server			
a] Logging	Configuration Bridging			
∲ Sparkplug	Enable Non-secure			
∽ MQTT Ω Alerts	Non-secure Port 💿	1883	3	
Diagnostics	Enable Secure 💿			
CONFIGURATION	Secure Port	8883	3	
<ul> <li>Accounts</li> <li>MOTT Credentials</li> </ul>	Enable WebSocket 💿			
MQTT Server	WebSocket Port	8090	3	
් License	Enable Secure WebSocket			
System	Secure WebSocket Port	8091	3	
+	Bind Address 💿	0.0.0.0		

The form contains the following fields:

- Enable Non-secure
- Whether to enable non-secure client connections over plain TCP.
- Non-secure Port
- The port that the MQTT Server will listen on for non-secure connections.
- Enable Secure
- Whether to enable Secure client connections over SSL/TLS.
- Secure Port

   The port that the MQTT Server will listen on for secure connections.
- Enable WebSocket
  - Whether to enable non-secure client connections over WebSockets.
- WebSocket Port
  - The port that the MQTT Server will listen on for non-secure WebSocket connections.
- Enable Secure WebSocket
  - Whether to enable secure client connections over WebSockets.
- Secure WebSocket Port
- The port that the MQTT Server will listen on for secure WebSocket connections.
- Bind Address
- The address that the MQTT Server will listen on for MQTT connections.
- Allow Anonymous
- Whether to allow anonymous connections (no username and password).
   Anonymous MQTT Credentials
  - The MQTT Credentials to use for anonymous client's ACLs

The "Update" button in the bottom right can be clicked to update the MQTT Server configuration once all changes have been made. An update will restart the MQTT server.

### Bridging

Λ

#### Warning

Note MQTT bridging is not compatible with Sparkplug. Bridging should only be used for non-Sparkplug related MQTT messages.

A Bridge Client allows you to connect Chariot to another MQTT Server to share messages between the two servers. The shared messages can be restricted by topic filters to only share a subset of messages flowing through either of the servers.

A common usage is to connect edge MQTT brokers to a central or remote broker.

When enabled on Chariot, the Bridge Client becomes an MQTT client to the other MQTT Server. This client publishes messages, flowing through Chariot, to the other broker. It also subscribes on topics with the other broker to receive messages and deliver them to subscribed clients connected to Chariot.

A	Bridging is supported in release 2.3.0 forward	
---	--	--

🔴 🕒 💿 👘 Chariot	× +				
$\leftrightarrow$ $\rightarrow$ $\mathbf{G}$	) 掐 ⊶ 192.168.1.81:	8080/#/mqttserver		☆	ල ⊻ ම 🕮 එ 📫
Chariot admin	МОТТ	Server ®			
STATUS					
📲 Dashboard	Home > MQTT Serv	er			
al Logging	Configuration	Bridging			
🖇 Sparkplug	Add Bridge Clier	ıt			
∖ MQTT	NAME	ADDRESS	PORT	CLIENT ID	ENABLED
û Alerts	b1	chariot-testing.chariot.io	8883	BRIDGE_02	Enabled
Diagnostics					
CONFIGURATION			( )		
器 Accounts					
MQTT Credentials					
MQTT Server					
🕑 License					
□ System					
<i>←</i>					

To edit an existing bridge client, select the client from the UI. To add a new bridge client , select Add Bridge Client.

🗧 🔵 🔹 📝 Chariot					
$\leftarrow \rightarrow \mathbf{C}$	0 🗟 👓	192.168.1.81:8080/#/mqttserver		☆	ල ⊻ ම 🗆 එ 📫
Chariot admin ▽	9	Add Bridge Clientunde	fined		
STATUS					
🗄 Dashboard	Home >	Name 💿			
al Logging	Conf	Enabled 📀			
🖇 Sparkplug	Add	Username 📀	admin		
∖ мqтт	NAME	Password 💿		8	ENABLED
û Alerts		Address 💿	localhost		
Diagnostics					
CONFIGURATION		Port 🕜	1883	$\hat{\cdot}$	
絕 Accounts		Use TLS 💿			
MQTT Credentials		Client ID 💿			
MQTT Server					
ඵ License		Keep Alive 💿	30	¢	
□ System		Clean Session 📀			
←		Allow Retained 💿			

The form contains the following fields:

- Name
- A unique name for the bridge connection.
- Enabled • Sets the enable state of the bridge connection.
- Username

   Optional username for connecting to the server.
- Address
  - The server address.
- Port
- The server port.
  - Whether the bridge connection will be using SSL/TLS.

By default Chariot comes with an empty truststore file clientcerts.jks which overrides the JVM cacerts truststore. If TLS is enabled, the following lines will need to be removed from the <Chariot\_install\_directory>/conf/com.cirruslink.chariot.system config file:

- trustStoreFile="security/clientcerts.jks"
- trustStorePassword="secretpassword"

If the signer of the SSL/TLS cert installed on the remote MQTT Server is an external, commercial CA (e.g., DigiCert) the bridge client should successfully connect over TLS once you update the config file and restart Chariot.

If the signer of the SSL/TLS cert installed on the remote MQTT Server is an internal, non-commercial, you will need to add your Root CA cert to the JVMs cacerts trust store and restart Chariot.

- Client ID
- The MQTT client ID to use for connecting to the server.
- Keep Alive
  - The MQTT keep alive time in seconds.
- Clean Session
  - Whether to connect with a clean session.
- Allow Retained
  - Whether the retained message flag is allowed on messages published to the server.
- Try Private
  - $^{\circ}~$  Whether the client should attempt to indicate to the server that it is a bridge client.
  - This feature helps to detect message loopback, but is only supported by some MQTT servers and may need to be disabled in order to connect.
- Sub Topics

- ° The topics filters that will be subscribed on by the bridge client.
- The topic filters are of the form <topic>:<qos> or just <topic>.
- Pub Topics
  - The topics filters that will be published on by the bridge client. These are used to restrict/filter which messages (flowing through Chariot) are published to the remote server,
  - ° The topic filters are of the form <topic>:<qos> or just <topic>.

The "Update" button in the bottom right can be clicked to update the MQTT Server Bridging configuration once all changes have been made

# License

The License page allows for uploading, activating, and/or deactivating a Chariot License.

A detailed tutorial for licensing Chariot can be found here Licensing Procedure.

A If you have deployed Chariot through AWS Marketplace or Azure Marketplace, then no additional steps are required - your license is already installed and activated.



### **Upload License**

A Chariot License is represented by a text file which contains a license key. The file can either be dragged/dropped into the "Upload License" form, or the license text can be pasted directly into the form.

	9 192.168.1.81.8080/#/license		۲ ۲		ਾ⊶ ਹ`=
Chariot admin Up	load License	No Active License		st ×	art Trial License
ATUS )} Dashboard a) Logging ∳ Sparkplug MQTT	# Charlot MQTT Server Standard 2.x License (id: xxxxxxxxxxxx) 578bf731cd71bc43c70c349d8f7d9c9a25ba0f93312333ad7ad3600 bc714dbe8c357d957bc44405219a3 7364880aba5f59918e65011a17d95967cb90df2f49f34a8bec74a11 93cff8311c2ccce9570a0b57934fa500ac13ec0ba8c2397cl6579bc4480 ac2e5a54f819175f780e6658c6284446c7a5b68737bc2f6800b 37202f0db90357e8bc15afd1958bc24c593faffc63310823de3580 abcc5a76df18259467998379d26bb989f2c2edf171d62c8399dabb	ff899e Sdb450 9dc53 9414a 6f075 7f4a4 ff07adb			
Alerts	) Activate			Cancel OK	
ා Diagnostics	Online Activation				
NFIGURATION  Accounts  MQTT Credentials	<b>\$</b>	$\bigcirc$			
MQTT Server	Activate License	Deactivate License			
□ System					
	←				

The Activate toggle (in the Upload License form) can be used to auto-activate the license immediately after it is uploaded. This is a convenience to prevent the need to click Activate License after uploading a license.

# **Online Activation**

Online Activation allows the Chariot MQTT Server to remotely connect to the Chariot Licensing server for activation and deactivation of Chariot Licenses. When the "Online Activation" is selected, the License page appears with the following options:

••• Char	riot × +		
$\leftarrow \rightarrow C$	이 견 아 192.168.1.81:8080/#/license	☆	ල ± ⊛ ූ දා ≓
Chariot admin	No Active License		Start Trial License
STATUS	License ©		
<ul> <li>✓ Sparkplug</li> <li>MQTT</li> <li>△ Alerts</li> </ul>	Status: INACTIVE License ID: 1642023997036 Activation Date: May 17, 2024 12:31 PM Activation Period: 0 Expiration: October 27, 2295 3:46 PM	L Upload License	
Diagnostics	Online Activation		
enfiguration 終 Accounts		2	
MQTT Credentials	Activate License Deactiva		
MQTT Server			
🕑 License			
System			
→ →			

- Activate License
- Activates an uploaded Chariot License by remotely accessing the Chariot Licensing Server.
- Deactivate License
  - $^\circ~$  Deactivates an uploaded & active Chariot License by remotely accessing the Chariot Licensing Server.

### **Offline Activation**

Offline Activation requires the user to generate an activation request and provide it to Cirrus Link Solutions in order to obtain an Activation Code. The Activation Code can then be uploaded into the Chariot MQTT Server to activate the previously uploaded Chariot License. When the "Offline Activation" option is selected, the License page appears with the following options:

🗕 🌢 💼 🍼 Chari	ot × +			~
$\leftarrow \rightarrow G$	○ A ~ 192.168.1.81:8080/#/license		☆	ତ ⊻ © ี £ =
Chariot admin		No Active License		Start Trial License
STATUS	License ⊙ Home > License			
∞ Sparkplug	Status: INACTIVE License ID: 1642023997036 Activation Date: May 17, 202 Activation Baried: 0.0	14 12:31 PM	<u>1</u>	
û Alerts	Expiration: October 27, 2295	3:46 PM Up		
Diagnostics	Offline Activation			
CONFIGURATION				
器 Accounts		<>		
MQTT Credentials	Generate Activation Request	Upload Activation Code	Deactiva	te and Generate
MQTT Server			Dodotina	Request
🔿 License				
□ System				

- Generate Activation Code
  - Generates an Activation Request Code.
  - Deactivate and Generate Code
    - Deactivates an active Chariot License and generates a Deactivation Request Code.
- Upload Activation Code
  - ° Uploads an Activation Code that was obtained using the Generate Activation Code.

# System

٠

The System page allows for the configuration of the Chariot MQTT Server's system settings as well as enabling secure (SSL/TLS) connections and uploading certificates.

A detailed tutorial for setting up SSL/TLS can be found here: Securing Chariot® MQTT Server.

It comprises three tabs: Configuration, Certificates and Backup/Restore

### Configuration

The System page uses the following form to configure the system settings:

Chariot	× +		
$\leftarrow$ $\rightarrow$ C $\textcircled{a}$	O D localhost:8080/#/system		⊗ <u>⊀</u> O ≓
Chariot admin マ STATUS	System © G <sup>o</sup> SSL Not Setup Home > System		
al Logging	Configuration Certificates Ba	ckup/Restore	
∲ Sparkplug	Server Name 💿	ExampleChariot	
Alerts	Enable HTTP 💿		
📓 Diagnostics	HTTP Port	8080	٢
CONFIGURATION	Enable HTTPS 💿		
<ul><li></li></ul>	HTTPS Port 💿	8443	0
MQTT Server			Update
🕘 License			
□ System			
+			

The form contains the following fields:

- Server Name
  - An optional name for this Chariot installation.
- Enable HTTP
  - Whether to enable non-secure HTTP connections to the Chariot Web UI.
- HTTP Port • The port that the Chariot MQTT Server will use for HTTP.
- Enable HTTPS
- Whether to enable Secure HTTPS connections to the Chariot Web UI.
   HTTPS Port
  - The port that the Chariot MQTT Server will use for HTTPS.

### Certificates

The certificates tab provides the means to upload the necessary certificates and keys for setting up SSL/TLS.

() Self-signed certificates should not be used in a production environment on a public network.

• • • Øß Chariot	× +							
$\leftarrow$ $\rightarrow$ C $\textcircled{a}$	🗘 🗅 locali	nost:8080/#/system			2	${igsidential}$	⊻ (	ວ ≓
Chariot admin ▽	Syste	em @						
STATUS		not octup						
[]] Dashboard	Home > System							
al Logging	Configuration	Certificates Backup/Restore						
🖗 Sparkplug	Upload the fol	lowing files and then click "Setup SSL						
∖ мотт	STATUS	FILE TYPE	NAME	TIME				
û Alerts	$\otimes$	Private Key 💿				1		
ii Diagnostics	$\otimes$	CA Chain 📀				1		
CONFIGURATION	$\otimes$	Certificate 📀				1		
器 Accounts								
MQTT Credentials						Se	tup SSL	
MQTT Server								
් License								
□ System								

The Certificates tab contains the following fields:

- Private Key

   An RSA private key of type PKCS1 in PEM format that was used to generate the certificate signing request for the server Certificate
   Review How to identify my Private Key type to verify the type
- CA Chain
  - An X.509 public root CA (Certificate Authority) certificate and any/all public intermediate CA certificates between the root and the CA that issued the certificate in PEM format. If there are no intermediate CAs, then the chain is made up of only the public root CA certificate.
- Certificate An X.509 server certificate in PEM format.

Once the three files are uploaded, the "Setup SSL" button can be clicked to finish the setup.

If successful, the certificate information will now be displayed on the Certificates tab:

Charlot   Status   Status <th>Chariot</th> <th>× +</th> <th></th> <th></th> <th></th> <th></th>	Chariot	× +				
Chaidid   Status   Status </th <th><math>\leftarrow</math> <math>\rightarrow</math> C <math>\textcircled{a}</math></th> <th>O 🗅 localhost:8080/#/system</th> <th></th> <th>公</th> <th>⊠ ⊻</th> <th>0 =</th>	$\leftarrow$ $\rightarrow$ C $\textcircled{a}$	O 🗅 localhost:8080/#/system		公	⊠ ⊻	0 =
Bashboard   al Logging   Sparkplug   Configuration   Certificates   Backup/Restore   Click "Delete" to remove the SSL certificates   SSL Details   Issue:: CN=CLS Testing Intermediate CA, OU=Support, O=Cirrus Link Solutions, ST=Kansas, C=US   Valid To:: August 13, 2028 535 PM   Subject: CN=Iocalhost, OU=Support, O=Cirrus Link Solutions, L=Spring Hill, ST=Kansas, C=US   Valid To:: August 13, 2028 535 PM   Subject: CN=Iocalhost, OU=Support, O=Cirrus Link Solutions, L=Spring Hill, ST=Kansas, C=US   Valid To:: August 13, 2028 535 PM   Version: 3   Diagnostics   CONFIGURATION   & Accounts   MQTT Credentials   MQTT Server   O License	Charlot admin STATUS	System © A SSL is Setup SSL successfully setup	×			
<ul> <li>Sparkplug</li> <li>Click "Delete" to remove the SSL certificates</li> <li>MQTT</li> <li>Alerts</li> <li>Diagnostics</li> <li>SSL Files</li> <li>Private Key: localhost.keypem</li> <li>Ca Chain: ca-chain.cert.pem</li> <li>MQTT Credentials</li> <li>MQTT Server</li> <li>License</li> <li>System</li> </ul>	旧 Dashboard ๗ Logging	Configuration Certificates Backup/Restore				
<ul> <li>Diagnostics</li> <li>SSL Files</li> <li>Private Key: localhost.key.pem</li> <li>ConFiguration</li> <li>Accounts</li> <li>MQTT Credentials</li> <li>MQTT Server</li> <li>License</li> <li>System</li> </ul>	<ul> <li>✓ Sparkplug</li> <li>MQTT</li> <li>Q Alerts</li> </ul>	Click "Delete" to remove the SSL certificates SSL Details Issuer: CN=CLS Testing Intermediate CA, OU=Support, O=Cirrus Link Solutions Signature Algorithm: SHA256withRSA Subject: Okalorationat OLaStronort O=Cirrus Link Solutions L=Spring Hill ST	s, ST=Kansas, C=US	Valid From: May 8, 2018 5:35 PM Valid To: August 13, 2028 5:35 PM Version: 3		
<ul> <li>MQTT Credentials</li> <li>MQTT Server</li> <li>License</li> <li>System</li> </ul>	Diagnostics CONFIGURATION ACCOUNTS	SSL Files Private Key: localhost.key.pern CA Chain: ca-chain.cert.pern Certificate: localhost.cert.pern				
MQTT Server  License  System	MQTT Credentials				Delete S	SL
	MQTT Server     License     System					

The "Delete SSL" button can be used to remove the certificate and key from the Chariot MQTT Server and revert to a "SSL Not Setup" state.

## Backup/Restore

This tab allows for the backup and restore of Chariot configuration data.

Chariot	× +			
$\leftarrow$ $\rightarrow$ C $\textcircled{a}$	O 🗅 localhost:8080/#/system	කි. ක	⊗ <u>⊻</u> O ≓	=
Chariot admin  ▽	System ©			
STATUS				
[]= Dashboard	ome > System			
₁d] Logging	Configuration Certificates Backup/Restore			
🖇 Sparkplug				
∖ мо∏	_↓_	<u>↑</u>		
û Alerts	Export Backup	Restore From Backup		
Diagnostics				
CONFIGURATION				
& Accounts				
MQTT Credentials				
MQTT Server				
🗇 License				
🗆 System				

- Export Backup
- This button downloads a backup.zip file that contains the current configuration data of the Chariot instance
   Restore From Backup
  - This button restore configuration data from backup.zip file.

The backup does not include any SSL certificates that may have been loaded. If configured for SSL, make sure to also copy the ssl.jks keystore from <OLD\_chariot\_install\_folder>/security directory and save the ssl.jks keystore and exported backup.tgz file in a safe, known location