Setting up Client based authentication with Client Certificates

This document describes how to configure Client based authentication with Client Certificates to create secure connections between Chariot, MQTT Engine and MQTT Transmission.

Available in Chariot version 2.3.1 and greater

- Prerequisites
 - Generating Certificates and Keys
 - Generate Root CA certificate
 - Generate MQTT Engine Client certificate signed with the Root CA's private key
 - Generate MQTT Transmission Client certificate signed with the Root CA's private key
- Setting up SSL Connections Using Two-way Authentication
 - Server Side Configuration
 - Setup SSL on Chariot
 - Update Chariot Truststore
 - Update Chariot Clients Authentication Policy
 - Client Side Configuration
 - MQTT Engine Client Side Configuration
 - MQTT Transmission Client Side Configuration
 - Anonymous Client Connections
- Verifying Connectivity
- Engine
 - Transmission
 - Chariot

Prerequisites

 \oslash

- Ignition installed with MQTT Engine and MQTT Transmission modules
- Chariot MQTT Server v2.3.1 or greater installed

The command line tools openssl and keytool are used.

Install the OpenSSL command line tool and add the OpenSSL PATH in the Windows environment variables if necessary.

Keytool is part of the standard java distribution and is located in the bin sub-directory of your jdk installation directory. Chariot includes a java distribution under the <chariot_install_dir>/lib/runtime/jdk11.0.12_7/bin folder. Add the keytool PATH in the Windows environment variables if necessary.

You will need to restart your any open command window to pick up this configuration change.

Generating Certificates and Keys

As a first step, we need to generate the certificate hierarchy for Chariot, MQTT Engine and MQTT Transmission.

Create the following folder structure on your local drive to hold the various certificates in the hierarchy that we will be generating:

```
chariotcerts/
ca/
certs/
engine/
transmission/
```

When creating a certificate hierarchy, the Root CA is the highest level of authority in the certificate hierarchy, and is responsible for issuing signed certificates, such as the MQTT Engine and MQTT Transmission certificates that will be shown below. When the Root CA issues a certificate, it signs the certificate with its private key, which allows the MQTT Server to verify the authenticity of the certificates using the Root CA's public key.

Note this tutorial only covers client based authentication using certificates. It does not cover setting up TLS/SSL on an MQTT Server which is used encyrpt communication between MQTT clients and the MQTT Server. This must also be done and information can be found here on how to set this up.

These are the steps that need to be completed for the certificate hierarchy:

- Generate Root CA
- Generate MQTT Engine Client certificate signed with the Root CA's private key

Generate MQTT Transmission Client certificate signed with Root CA's private key

Generate Root CA certificate

1. Generate a private key file (ca.key) for the Root CA using the command below. You may choose to enter a passphrase to be associated with the ca.key file as well.

Make note of this passphrase if you set one for the Root CA private key file (ca.key) as it will be used multiple times.

openssl genrsa -des3 -out ca/ca.key 4096

 Generate a self-signed certificate (ca.crt) for the Root CA using the command below. This command generates a new self-signed X.509 certificate named "ca.crt" valid for 3650 days (10 years) using the RSA private key "ca.key". You will be required to enter the pass phrase associated with the private key file "ca.key".

openssl req -new -x509 -key ca/ca.key -days 3650 -out ca/ca.crt

M There are a number of fields associated with the creation of the certificate. Fill them out with your relevant details.

Example CA Creation

```
$ openssl req -new -x509 -key ca/ca.key -days 3650 -out ca/ca.crt
Enter pass phrase for ca/ca.kev:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
____
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:KS
Locality Name (eg, city) []:Stilwell
Organization Name (eg, company) [Internet Widgits Pty Ltd]: Cirrus Link Solutions
Organizational Unit Name (eg, section) []:Support
Common Name (e.g. server FQDN or YOUR name) []:CLS Example Root CA
Email Address []:
Ś
```

You should have the following files created:

chariotcerts/ ca/ ca.crt ca.key

A Depending on the version of openSSL that you are using, you may see additional .srl files created which contain the signed certificate's unique serial number. These files are not used directly by the modules and not included in the certificate hierachy displayed above.

Generate MQTT Engine Client certificate signed with the Root CA's private key

1. Generate private key in PSCK8 format (engine.key) for MQTT Engine using the command below.

openssl genrsa -out certs/engine/engine.key 4096

Generate a Certificate Signing Request (CSR) for MQTT Engine using the command below. This command generates a new CSR named "engine. csr' using the RSA private key "engine.key". openssl req -new -key certs/engine/engine.key -out certs/engine/engine.csr

M There are a number of fields associated with the creation of the certificate. Fill them out with your relevant details.

Example Engine CSR Creation

\$ openssl req -new -key certs/engine/engine.key -out certs/engine/engine.csr You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank For some fields there will be a default value, If you enter '.', the field will be left blank. Country Name (2 letter code) [AU]:US State or Province Name (full name) [Some-State]:KS Locality Name (eg, city) []:Stilwell Organization Name (eg, company) [Internet Widgits Pty Ltd]:Cirrus Link Solutions Organizational Unit Name (eg, section) []:Support Common Name (e.g. server FQDN or YOUR name) []:EngineDevice Email Address []: Please enter the following 'extra' attributes to be sent with your certificate request A challenge password []: An optional company name []: \$

3. Sign the MQTT Engine Client CSR with the Engine CA using the command below. This command will sign the CSR "engine.csr" with the Root CA certificate 'ca.crt' and Root CA's RSA private key 'ca.key', creating a new X.509 certificate named 'engine.crt' valid for 365 days (1 year). You will be required to enter the passphrase associated with the private key file "ca.key".

openssl x509 -req -in certs/engine/engine.csr -CA ca/ca.crt -CAkey ca/ca.key -CAcreateserial -out certs /engine.crt -days 365

Generate MQTT Transmission Client certificate signed with the Root CA's private key

1. Generate private key in PKCS8 format (transmission.key) for MQTT Transmission using the command below.

openssl genrsa -out certs/transmission/transmission.key 4096

 Generate a Certificate Signing Request (CSR) for MQTT Transmission using the command below. This command generates a new CSR named "transmission.csr' using the RSA private key "transmission.key".

openssl req -new -key certs/transmission/transmission.key -out certs/transmission/transmission.csr

A There are a number of fields associated with the creation of the certificate. Fill them out with your relevant details.

Example Transmission CSR Creation

\$ openssl reg -new -key certs/transmission/transmission.key -out certs/transmission/transmission.csr You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank For some fields there will be a default value, If you enter '.', the field will be left blank. Country Name (2 letter code) [AU]:US State or Province Name (full name) [Some-State]:KS Locality Name (eg, city) []:Stilwell Organization Name (eg, company) [Internet Widgits Pty Ltd]: Cirrus Link Solutions Organizational Unit Name (eg, section) []:Support Common Name (e.g. server FODN or YOUR name) []:TransmissionDevice1 Email Address []: Please enter the following 'extra' attributes to be sent with your certificate request A challenge password []: An optional company name []: \$

3. Sign the MQTT Transmission Client CSR with the Transmission CA using the command below. This command will sign the CSR "transmission. csr" with the Root CA certificate 'ca.crt' and Root CA's RSA private key 'ca.key', creating a new X.509 certificate named 'transmission.crt' valid for 365 days (1 year). You will be required to enter the passphrase associated with the private key file "transmission.key".

```
openssl x509 -req -in certs/transmission/transmission.csr -CA ca/ca.crt -CAkey ca/ca.key -CAcreateserial -out certs/transmission/transmission.crt -days 365
```

We have now generated all the certificates and keys needed to setup MQTT Client certificate based authentication connections between Chariot and the MQTT Engine and MQTT Transmission modules:

chariotcerts/ ca/ ca.crt ca.key certs/ engine.crt engine.csr engine.key transmission/ transmission.csr transmission.key

Setting up SSL Connections Using Two-way Authentication

Now we are ready to setup SSL connections between two clients (MQTT Engine and Transmission) and the Chariot Server.

Here is a summary of what needs to be done:

- Server side configuration
 - · Enable SSL on Chariot and add server side certificates and keys (cachain.crt, private.key, and servercertificate.crt)
 - ° Add Client certificates (engine.crt and transmission.crt) to the Chariot truststore
 - Set the 'Clients Authentication Policy' on Chariot to "required"
- Client side configuration
 - Add the engine.key engine.crt to the 'Chariot' connection on the MQTT Engine side.
 - Add the transmission.key and transmission.crt to the 'Chariot' connection on the MQTT Transmission side.

Server Side Configuration

Setup SSL on Chariot

Navigate to CONFIGURATION > System > Certificates configuration and upload the files as shown below. Once uploaded, select the Setup SSL button. Use the certificate components created in Secure Chariot MQTT Server communication using SSL/TLS.

File Type	Where to get the file
CA Chain	Provided by your Certificate Authority
Private Key	The key you generated when creating your CSR to submit to your CA
Certificate	The server certificate provided by your Certificate Authority after you submitted your CSR to them

Chariot	× +			,	~
$\leftarrow \rightarrow $ G	C & or chariot-testing.chariot.io:8080/#/system		ដ	♥ @ ;	മ ≡
Chariot admin ▽	System ©				
STATUS					
[]= Dashboard	Home > System				
₁d] Logging	Configuration Certificates Backup	Restore			
🕸 Sparkplug	Upload the following files and then click "	Setup SSL"			
∖ мqπ	STATUS FILE TYPE	NAME	TIME		
ධ Alerts	Private Key	chariot.io.key	April 30, 2024 3:40 PM	1	
Diagnostics	CA Chain 💿	cachain.pem	April 30, 2024 3:40 PM	Î	
CONFIGURATION	Certificate	chariot_io.pem	April 30, 2024 3:41 PM		
器 Accounts					
MQTT Credentials				Setup SSL	
MQTT Server					
🔿 License					
🗆 System					
←					



Navigate to CONFIGURATION > MQTT Server configuration and Enable Secure as shown below. Select the Update button to save the configuration.

						·
← → C	O A https://chariot-certs.chariot.io/#/mqttserver	☆	8) 🗄	Ħ	ර
Chariot admin ▽	MQTT Server ©					
TATUS	Ŭ					
📲 Dashboard	Home > MQTT Server					
₀d Logging	Configuration Bridging					
🖇 Sparkplug	Enable Man conjunc					
💊 мотт						
û Alerts	Non-secure Port 1883		\$			
Diagnostics	Enable Secure					
ONFIGURATION	Secure Port 8883		0			
器 Accounts						

Update Chariot Truststore

By default Chariot comes with an empty truststore file clientcerts.jks which is located in the <chariot_install_dir>/security folder.

To view this file, run the command as shown below. You will be required to enter the keystore password and this can be found in the <chariot_install_dir>/conf/com.cirruslink.chariot.system configuration file as the "trustStorePassword" parameter.

```
keytool -list -v -keystore <chariot_install_dir>/security/clientcerts.jks
```

You will see that the truststore contains no entries.

Use the following command to add the Root CA certificate to the truststore using the "trustStorePassword" when prompted.

When prompted Trust this certificate? [no]: respond "yes"

keytool -importcert -file ca/ca.crt -keystore <chariot_install_dir>/security/clientcerts.jks -alias CACertificate

Use the following command to add the Engine client side certificate to the truststore using the "trustStorePassword" when prompted.

When prompted Trust this certificate? [no]: respond "yes"

```
keytool -importcert -file certs/engine/engine.crt -keystore <chariot_install_dir>/security/clientcerts.jks -
alias EngineDevice
```

Use the following command to add the Transmission client side certificate to the truststore using the "trustStorePassword" when prompted.

When prompted Trust this certificate? [no]: respond "yes"

```
keytool -importcert -file certs/transmission/transmission.crt -keystore <chariot_install_dir>/security
/clientcerts.jks -alias TransmissionDevicel
```

Once completed, viewing the file will now show three entries similar to below:

```
Keystore type: PKCS12
Keystore provider: SUN
Your keystore contains 3 entries
Alias name: cacertificate
Creation date: Feb 7, 2024
Entry type: trustedCertEntry
Owner: CN=CLS Example Root CA, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Issuer: CN=CLS Example Root CA, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Serial number: 6f689c58c0f3e7177224c5868b75fcd51bcc2e0f
Valid from: Wed Feb 07 19:43:45 UTC 2024 until: Sat Feb 04 19:43:45 UTC 2034
Certificate fingerprints:
        SHA1: 15:BC:A7:28:BE:15:D9:7F:E7:B0:1E:51:4E:A0:0F:57:58:C7:A3:2E
        SHA256: 7A:1E:A0:D8:B5:4E:CB:BF:2D:A9:8D:E2:7F:E6:20:3C:B8:2C:11:4F:14:FF:AD:F6:A1:01:58:C0:37:B3:04:A7
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3
*****
Alias name: enginedevice
Creation date: Feb 7, 2024
Entry type: trustedCertEntry
Owner: CN=EngineDevice, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Issuer: CN=CLS Example Root CA, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Serial number: 6f689c58c0f3e7177224c5868b75fcd51bcc2e0f
Valid from: Wed Feb 07 19:43:45 UTC 2024 until: Sat Feb 04 19:43:45 UTC 2034
Certificate fingerprints:
        SHA1: 15:BC:A7:28:BE:15:D9:7F:E7:B0:1E:51:4E:A0:0F:57:58:C7:A3:2E
        SHA256: 7A:1E:A0:D8:B5:4E:CB:BF:2D:A9:8D:E2:7F:E6:20:3C:B8:2C:11:4F:14:FF:AD:F6:A1:01:58:C0:37:B3:04:A7
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3
*****
Alias name: transmissiondevice1
Creation date: Feb 7, 2024
Entry type: trustedCertEntry
Owner: CN=TransmissionDevice1, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Issuer: CN=CLS Example Root CA, OU=Support, O=Cirrus Link Solutions, L=Stilwell, ST=KS, C=US
Serial number: 6f689c58c0f3e7177224c5868b75fcd51bcc2e11
Valid from: Wed Feb 07 19:52:48 UTC 2024 until: Thu Feb 06 19:52:48 UTC 2025
Certificate fingerprints:
        SHA1: 60:7E:A5:6E:47:09:79:FB:A9:FE:24:DB:05:1E:09:54:24:48:19:BD
       SHA256: 8A:C8:39:E1:50:8B:BF:35:25:43:C7:B4:66:60:02:1E:AF:4F:C4:11:32:B0:6D:FC:6D:6E:5D:A8:BE:FA:00:0B
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 4096-bit RSA key
Version: 3
*****
********
```

Update Chariot Clients Authentication Policy

Using a text editor, set the "clientAuthPolicy" to "required" in the <chariot_install_dir>/conf/com.cirruslink.chariot.server configuration file.

```
clientAuthPolicy="required"
```

You will now need to restart the Chariot service to pickup up the configuration changes

Client Side Configuration

MQTT Engine Client Side Configuration

Add the certificates to the MQTT Engine > Servers > Certificates configuration as shown below:

Friendly Name	Certificate Filename	File Description	File Location
EngineCertificate	engine.crt	MQTT Engine Certificate	chariotcerts/certs/engine
EngineKey	engine.key	MQTT Engine Private Key	chariotcarts/certs/engine

tion				🕹 admin Log C
nition				Help 🛛 Get Designe
SYSTEM	Config > Mqttengine > MQTT Engin	e Settings		
Overview	rial Mode 1:22:44 We're glad you're t	est driving our software. Have fun.		Activate Igni
Backup/Restore				
Licensing	General Servers	Namespaces		
Modules				
Projects	Settings Certifica	tes		
Gateway Settings				
	Friendly Name	Certificate Filename	File Description	
NETWORKING	Engine Certificate	engine.crt	MQTT Engine Certificate	delete
Email Settings		-		
Gateway Network	Engine Key	engine.key	MQTT Engine Private Key	delete
SECURITY	→ Create new Certificate	2		
General				
Auditing	Note: For additional detail	s on configuring MQTT Engine, see the		
Users, Roles	documentation here			
Identity Providers				
OAuth2 Clients				
Security Levels				

Update the MQTT Engine > Servers > Settings configuration to use the certificates as shown below and setting the URL to be ssl://FQDN:8883 with the FQDN of the Chariot Server. Click the Save Changes button to save the configuration. Note the URL must use SSL. If a non-secure connection is specified here, the connection will not succeed.

						_				
\rightarrow C	() 🗋 localho	ost:8088/web/config/r	nqttengine.settings?62		${\bigcirc}$	⊻	쁐	٤	
We	b Server	🌣 Config 🗦	Mqttengine > MQTT I	ingine Settings						
ne Em	ail Settings	Trial Mod	e 1:11:33 We're glad y	ou're test driving our software. Have fun.				Activat	e Igniti	ion
	leway Network		Fachled							
US SECUR	ITY		Enabled	Enable this MQTT Server Setting						
Ger	neral			ssl://FODN:8883						
ig Aud	diting		URL	The URL of the MQTT Server to connect to. Should be of the form tcp://mydomain.com:18	83 or ssl://mydomain.com:88	83				
Use	ers, Roles							_		
Ser	vice Security		Username	admin						
Ide	ntity Providers			The username for connections if required by the MQTT Server (optional)						
Sec	curity Levels		Change							
Sec	curity Zones		Password?	Check this box to change the existing password.						
1										
DATAB	ASES		Password	The persuard for connections if required by the MOTT Server (optional)						
Cor	nnections			The password for connections in required by the most riserver (optional)				_		
Driv	vers									
Sto	re and Forward		Password	Re-type password for verification.						
ALARM	IING									
Ger	neral									
Jou	ırnal		TLS							
Not	tification		CA Cartificate File	- none - 🔻						
On	-Call Rosters		CA Certificate File	CA Certificate file currently in use						
Sch	nedules							_		
TAGS			Client Certificate	Engine Certificate v						
His	tory		THC .	Client certificate file currently in use						
_			Client Drivete Y	Facine Value						
Q			File	Client private key file currently in use						

MQTT Transmission Client Side Configuration

Add certificates to the MQTT Transmission > Servers > Certificates configuration as shown below:

Friendly Name	Certificate Filename	File Description	File Location
TransmissionCertificate	transmission.crt	MQTT Transmission Certificate	chariotcerts/certs/transmisson
TransmissionKey	transmission.key	MQTT Transmission Private Key	chariotcerts/certs/transmission

tion				-			_	👗 admin	Log Ou
nition								Help 🛛 Get I	Designer
SYSTEM	🌣 Config 🗲 Mqtttransm	ission > MQTT	Transmissio	n Settings					
Overview	Trial Mode 0:59:25	We're glad you're	test driving our	software. Have fun.				Act	ivate Ignitic
Backup/Restore									
Ignition Exchange	General	Servers	Sets	Transmitters	Records	Files			
Licensing									
Projects									
Redundancy	Settings	Certifica	ates						
Gateway Settings									
NETWORKING	Friendly	Name		Certificate Filen	ame	File Descript	tion		
Web Server	Transm	ssion Certificat	e	transmission.crt	transmission.crt MQTT Transmission Certificate			delete edit	
Email Settings	Turner	ssion Kov		transmission kou	MOTT Transmission Private Kou				
Gateway Network	Transm	ission key		transmission.key	transmission.key MQ11 Transmission Private Key			delete	
SECURITY	→ Create	new Certificat	te						
General									
Auditing	Note: For	additional detai	ils on configu	ring MQTT Transmissi	on, see the				
Users, Roles	documen	tation here							
Identity Providers									
OAuth2 Clients									
Security Levels									
Security Zones									

Update the MQTT Transmission > Servers > Settings configuration to use the certificates as shown below. Click the Save Changes button to save the configuration. Note the URL must use SSL. If a non-secure connection is specified here, the connection will not succeed.

Ignition - Ig	gnition Gateway × +						\sim
\rightarrow G (localhost:8088/web/config/n	nqtttransmission.settings?89	4	${\times}$	\pm	ញ	ර
Email Settings	Config > Mqtttransmission > Mq	QTT Transmission Settings					
Gateway Network	Trial Mode 0:57:51 We're glad y	ou're test driving our software. Have fun.			Ac	ctivate (gnition
s security General	URL	ssl://FQDN:8883 The URL of the MQTT Server to connect to. Should be of the form tcp://mydomai	n.com:1883 or ssl://mydomain.	com:8883			
g Auditing Users, Roles	Enabled	Enable this MQTT Server connection					
Service Security Identity Providers OAuth2 Clients	Server Set	Default * The Server Set this MQTT Server is associated with					
Security Levels Security Zones	Username	admin The username for this MQTT connection if required by the MQTT Server (optiona)				
DATABASES Connections	Change Password?	Check this box to change the existing password.					
Drivers Store and Forward	Password	The password for this MQTT connection if required by the MQTT Server (optional)				
ALARMING General Journal	Password	Re-type password for verification.					
On-Call Rosters	TLS						
TAGS	CA Certificate File	- none - CA Certificate file currently in use					
Realtime	Client Certificate File	Transmission Certificate Client certificate file currently in use					
Q Search	Client Private Key File	Transmission Key				-	

Anonymous Client Connections

Chariot MQTT Server still requires MQTT credentials to authenticate incoming client connections, even when using client certificates to set up a TLS/SSL session. If clients will not be sending an MQTT username and password, anonymous connections must be enabled.

To enable anonymous connections, navigate to the Configuration MQTT Server Configuration tab and set Allow Anonymous

$- \rightarrow \times$	○ À 192.168.1.81:8080/#/mqttserve	r -	☆	⊘ ີ ປີ ≐
ius Chariot Io admin	Enable Secure			
ATUS	Secure Port	8883		٢
] : Dashboard ଜା Logging	Enable WebSocket			
∯ Sparkplug	WebSocket Port	8090		٢
η μάιτ	Enable Secure WebSocket 💿			
Ĵ Alerts	Secure WebSocket Port	8091		\$
NFIGURATION	Bind Address 💿	0.0.0.0		
🌣 Roles	Allow Anonymous 📀			
MQTT Credentials	s Anonymous MQTT Credentials	None Selected		-
MQTT Server				
ා License				Update

By default, an anonymous client connection will be allowed to publish and subscribe on # unless the Anonymous MQTT Credentials has been selected.

This will allow you to select any of the configured MQTT Credentials, configured under Configuration MQTT Credentials, and MQTT Chariot will use the Publish and Subscribe ACLs for that MQTT Credential for all anonymous connections.

(i) A Password will need to be configured for this MQTT Credential but will not be used by MQTT Chariot

Verifying Connectivity

Engine

From the left hand menu bar, navigate to Config > MQTT Engine > Servers and note the Status as Connected.

	0.0				
\rightarrow G	Iocalhost:8088/web/conf	g/mqttengine.settings?97		☆	© ⊻ ╨ ĭ
nition					Log Οι
nition					Help 🛛 Get Designer
SYSTEM	🌣 Config > Mqttengine > MQ	T Engine Settings			
0verview	Trial Mode 0:56:21 We'reg	ad you're test driving our software. Have fun.			Activate Igniti
Backup/Restore					
us Ignition Exchange	Coursel Co				
Licensing	General Se	vers Namespaces			
fig Modules					
Projects	Settings	Certificates			
Gateway Settings					
- Gateria) Gettings	Name	URL	Username	Status	
NETWORKING					
Web Server	Chariot SCADA	ssl://chariot-certs.chariot.io:8883	engine	Connected	delete
		OTT Server Setting			
Q Search	→ Create new M	QTT Server Setting			

Transmission

From the left hand menu bar, navigate to Config > MQTT Transmission > Servers and note the Status as "x of x". This denotes the number of configured transmitters that are connected.

If you do not see a transmitter connected, verify that you have a transmitter with a valid Sparkplug ID either through setting the Group and Edge ID or through the TagPath. Review our troubleshooting guide for assistance.

$\rightarrow C$		ocalhost-8088/web/c	onfia/mattt	ransmission	settings?112				5~7	\bigtriangledown	\downarrow	LT	ናጉ
Ignition		ocantost.0000,web/c	Johng/Indeed	18113111331011	.56tting31112	_	_		22	0	•	~	- 0
											adm	iin Lo	g Out
gnition										Help 🕜	Ge	t Desig	ner
SYSTEM	¢ 0	ionfig > Mqtttransmissio	n > MQTT	Transmissio	n Settings								
ome Overview	Tria	lMode 0:53:10 We	Mode 0:53:10 We're glad you're test driving our software. Have fun.										gnitior
Backup/Resto	e												
Ignition Excha	nge												
Licensing		General	Servers	Sets	Transmitters	Records	Files						
nfig Modules													
Projects		Settings	Certifica	ates									
Redundancy													
Gateway Setti	igs												
NETWORKING		 Success 	fully update	ed MQTT Ser	ver "Charlot SCADA"								
Web Server		Name		URL			Server Set	Username	Connected				
Email Settings		Chariat 600	DA	celu//eboriet	sorts chariat io.0002		Default	ongino	1 of 1	delete	odi		
Gateway Netw	ork	charlot SCA		ssu//undriot-	certs.ci1d1101.10.8883		Deldult	engine	1011	uelete	edi		

Chariot

On the Chariot MQTT server, navigate to STATUS > MQTT where the number of active MQTT Clients will be displayed. This will be a count of 2 or 3 depending on your MQTT Transmission RPC Client configuration.

• • • 🖻 🚿 Chari	ot × +				
$\leftarrow \rightarrow G$	O A https://chariot-certs.chariot.io/#/mqttstatus			☆	0
Chariot admin ▽	MQTT Status 💿				
STATUS	v				
📲 Dashboard	Home > MQTT Status				
ၿါ Logging	Clients Retained Messages				
🖇 Sparkplug	Active MOTT Clienter 2		-	Disconnect	
`ν ΜΩΤΤ	Active Might Cirents, 2			Disconnect	
û Alerts	Network	Session		Last Will	
Diagnostics	IP Address -	Client ID	-	Will Topic	-
CONFIGURATION	State -	Username	-	Will QoS	-
	Last Active	Clean Start		Will Retained	

Clicking on the drop down will show the IDs of each client along with additional details:

🔸 🌢 👘 🍼 Char	riot × +									\sim	,
$\leftarrow \rightarrow $ G	O A https://chariot-certs.chari			☆	\bigtriangledown	⊥ 1	් ර	J			
Chariot admin ⊽	M	QTT Status 💿									
	Linne > MOT	T Status									
🗄 Dashboard	Home > MQT	Status									
₀d Logging	Clients										
🖇 Sparkplug	Active MQTT Clients: 2				*	Disconnect	C Live				
∖ мqтт	ME-8d38	38ff2-3e4e-4e3f -									
ậ Alerts	MT-ele3	MT-ele3b9a8-49ea-469f - 73.96.100.174					Last Will				
Diagnostics	IP Addre	ss	-	Client ID	-	Will Topic					
	State		-	Username	-	Will QoS					
	Last Acti	1/0		Clean Start		Will Retained					