

# FAQ: Chariot MQTT Server

This document provides answers to frequently asked questions about the Chariot® MQTT Server.

- **General Questions**
  - Where can I download Chariot?
  - What is the recommended size of a VM or EC2 instance to run Chariot MQTT Server?
  - I'm upgrading my Chariot server hardware. Does this invalidate my existing license?
  - Why am I not seeing new features in the Chariot UI after upgrading?
  - What are the default credentials for the Chariot MQTT Server?
  - I have MQTT clients failing to connect over TLS/SSL with a "Handshake timeout, no data received" error. Can this SSL handshake timeout be increased?
  - What format of private key must I use?
  - How do I configure the Chariot max heap size?
  - How do I configure the Chariot maximum outbound queue size?
  - How do I manage log files?
  - Is bridging supported by Chariot?
  - I enabled TLS and my Bridge Client does not connect
  - I am getting an error "...yajsw\bat\..tmp file does not exist" when attempting to installing or upgrade Chariot
  - Chariot License Mechanisms
  - How do I add a Read Only User
- **AWS Marketplace Questions**
  - Where does Chariot save my sensitive information?
  - How does Chariot encrypt data?
  - How do I set up and rotate SSL certificates and keys?
  - How do I decrypt my encrypted data?
  - How do I monitor the health of my Chariot server?
- **Azure Marketplace Questions**
  - Failure to deploy with subscription error

## General Questions

### Where can I download Chariot?

You will find the latest Chariot downloads here: [Chariot® Downloads](#).

### What is the recommended size of a VM or EC2 instance to run Chariot MQTT Server?

The Chariot MQTT Server runs best using a contemporary system with at least 4 CPU cores and 8GB of memory.

### I'm upgrading my Chariot server hardware. Does this invalidate my existing license?

Yes. You will need to temporarily deactivate your license before shutting down the Chariot server pre-upgrade. Next perform the server upgrade. Finally, activate your existing license by re-uploading and reactivating as shown [here](#).

### Why am I not seeing new features in the Chariot UI after upgrading?

Clear your web browser cache after upgrading Chariot to pick up the latest Chariot UI features and updates.

### What are the default credentials for the Chariot MQTT Server?

Chariot Admin User (UI/REST)

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

**exception:** password will be the EC2 instance ID (e.g., i-0049ac1e13e558b70) if deployed via [AWS Marketplace](#)

Chariot MQTT Admin User

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

### I have MQTT clients failing to connect over TLS/SSL with a "Handshake timeout, no data received" error. Can this SSL handshake timeout be increased?

Yes. Currently this configuration option is not available in the UI. First ensure you're running a Chariot version of **2.1.1** with a build date of **2021 May 13th or later** (if not, upgrade). Next add the following line to the `<Chariot_Install_Dir>/conf/com.cirruslink.chariot.server.config` file.

```
sslHandshakeTimeout=I"30000"
```

This sets the timeout to 30000 ms (30 sec). Increase the timeout further if required.

## What format of private key must I use?

The Private Key must be a RSA private key of type PKCS1 or PKCS8 in PEM format. Review [How to identify my Private Key type](#) to verify the type



PKCS8 support was added in release 2.3.1

## How do I configure the Chariot max heap size?

The Chariot max heap size is not configured and in prior to V2.2.3 defaulted to 1GB. In V2.2.3 onward the max heap size defaults to 2GB.

The Chariot max heap size can be configured on all version of Chariot in the `<chariot_install_dir>/yajsw/conf/wrapper.conf` file.

```
# Example 4GB Max Heap 2
wrapper.java.additional.6 = -Xmx4g
```

Note: a [Chariot service restart](#) is required for Chariot to pickup the configuration change.

## How do I configure the Chariot maximum outbound queue size?

The Chariot maximum outbound queue size can be configured in the `<chariot_install_dir>/conf/com.cirruslink.chariot.server` configuration file.

```
# Example set max outbound queue size to 1000
maxOutboundQueueSize = I"1000"
```

## How do I manage log files?

The Chariot wrapper.log text files can be managed on all version of Chariot in the `<chariot_install_dir>/yajsw/conf/wrapper.conf` file.

```
wrapper.logfile.loglevel=INFO

wrapper.logfile=${wrapper_home}/../log/wrapper.log

wrapper.logfile.format=LPDTM

wrapper.logfile.maxsize=10m

wrapper.logfile.maxfiles=10

wrapper.logfile.rollmode=SIZE

wrapper.logfile.maxdays=60

wrapper.logfile.desc=true
```

Based on this, up to 10 files of 10MB is size each may be retained. However, this isn't exact and if logging is very verbose these can surpass that size limit and end up being larger.

Note: a [Chariot service restart](#) is required for Chariot to pickup the configuration change.

The DB logfile can be managed in the `com.cirruslink.chariot.persistence.h2-logging.config` file. In this is a config option:

```
tables.1.maxRows=I"262144"
```

When maxRows is set, there will actually be two tables of that size created. When one fills up it will truncate the other one, to clear all the data, and then switch to using it rotating between the two tables.

Note: a [Chariot service restart](#) is required for Chariot to pickup the configuration change.

## Is bridging supported by Chariot?

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.

Bridging is supported in version 2.3.0 forward. See the [Chariot MQTT Server Configuration](#) guide for details on configuring bridging.

## I enabled TLS and my Bridge Client does not connect

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.

Chariot's JVM cacerts trust store path is <chariot\_install\_folder>/lib/runtime/amazon-corretto-<java\_version\_and\_platform>/lib/security/cacerts

NOTE: password is 'changeit'

## I am getting an error "....yajsw\bat\..tmp file does not exist" when attempting to installing or upgrade Chariot

This error is because the install batch file is not being executed with Administrator privileges.

## Example log detail

```
PS C:\Program Files\Chariot\02\03\01> .\install.bat
"C:\Program Files\Chariot\02\03\01\install.bat": Installing Amazon Corretto
"C:\Program Files\Chariot\02\03\01\install.bat" Installing Chariot service
"java" -Xmx30m -Dwrapper_home="C:\Program Files\Chariot\02\03\01\yajsw\bat\.." -Djna_tmpdir="C:\Program
Files\Chariot\02\03\01\yajsw\bat\..\tmp" -Djava.net.preferIPv4Stack=true --add-opens=java.base/java.lang=ALL-
UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --
add-opens=java.base/java.io=ALL-UNNAMED -jar "C:\Program Files\Chariot\02\03\01\yajsw\bat\..\wrapper.jar" -i
"C:\Program Files\Chariot\02\03\01\yajsw\bat\..\conf\wrapper.conf"
YAJSW: yajsw-stable-12.14
OS : Windows Server 2019/10.0/amd64
JVM : Amazon.com Inc./11.0.12/C:\Program Files\Chariot\02\03\01\lib\runtime\jdk11.0.12_7/64
Apr 18, 2023 8:54:42 AM com.sun.jna.Native loadNativeDispatchLibrary
WARNING: JNA Warning: IOException removing temporary files
java.io.IOException: JNA temporary directory 'C:\Program Files\Chariot\02\03\01\yajsw\bat\..\tmp' does not
exist
at com.sun.jna.Native.getTempDir(Native.java:1313)
at com.sun.jna.Native.removeTemporaryFiles(Native.java:1323)
at com.sun.jna.Native.loadNativeDispatchLibrary(Native.java:919)
at com.sun.jna.Native.<clinit>(Native.java:195)
at org.rzo.yajsw.os.ms.win.w32.WindowsXPProcess$MyKernel32.<clinit>(WindowsXPProcess.java:305)
at org.rzo.yajsw.os.ms.win.w32.WindowsXPProcess.changeWorkingDir(WindowsXPProcess.java:4149)
at org.rzo.yajsw.os.ms.win.w32.OperatingSystemWindowsXP.setWorkingDir(OperatingSystemWindowsXP.java:131)
at org.rzo.yajsw.WrapperExe.main(WrapperExe.java:115)
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.base/java.lang.reflect.Method.invoke(Method.java:566)
at org.rzo.yajsw.boot.WrapperExeBooter.main(WrapperExeBooter.java:49)
java.lang.reflect.InvocationTargetException
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.base/java.lang.reflect.Method.invoke(Method.java:566)
at org.rzo.yajsw.boot.WrapperExeBooter.main(WrapperExeBooter.java:49)
Caused by: java.lang.UnsatisfiedLinkError: Failed to create temporary file for /com/sun/jna/win32-x86-64
/jnidispatch.dll library: JNA temporary directory 'C:\Program Files\Chariot\02\03\01\yajsw\bat\..\tmp' does not
exist
at com.sun.jna.Native.loadNativeDispatchLibraryFromClasspath(Native.java:1032)
at com.sun.jna.Native.loadNativeDispatchLibrary(Native.java:988)
at com.sun.jna.Native.<clinit>(Native.java:195)
at org.rzo.yajsw.os.ms.win.w32.WindowsXPProcess$MyKernel32.<clinit>(WindowsXPProcess.java:305)
at org.rzo.yajsw.os.ms.win.w32.WindowsXPProcess.changeWorkingDir(WindowsXPProcess.java:4149)
at org.rzo.yajsw.os.ms.win.w32.OperatingSystemWindowsXP.setWorkingDir(OperatingSystemWindowsXP.java:131)
at org.rzo.yajsw.WrapperExe.main(WrapperExe.java:115)
... 5 more
"C:\Program Files\Chariot\02\03\01\install.bat": Current Chariot status: NOT_INSTALLED
PS C:\Program Files\Chariot\02\03\01>
```

## Chariot License Mechanisms

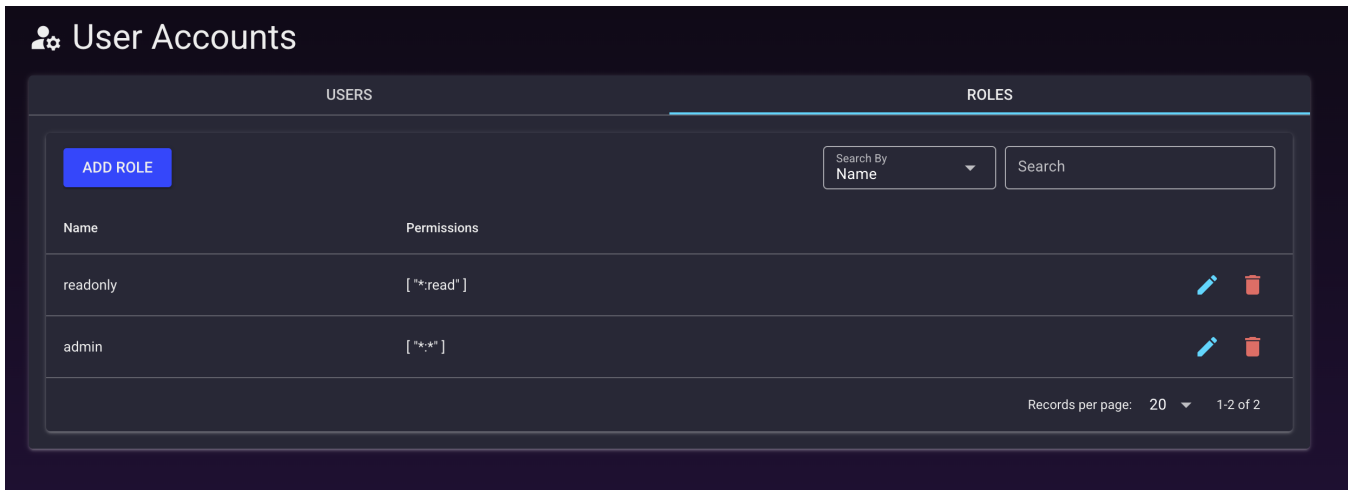
If you deploy the Chariot MQTT Server via our Cloud Marketplace offerings for Azure and AWS, the licensing/billing is rolled into your Azure/AWS bill. When you deploy the Chariot MQTT Server via Azure/AWS, you will not need to install a license of any kind – its licensed automatically.

If you are not deploying via our Cloud Marketplace offering, a one time perpetual license for Chariot is available and this license will be good forever. Contact [sales@cirrus-link.com](mailto:sales@cirrus-link.com) for pricing.

## How do I add a Read Only User

A users permissions have the form `<domain>:<action>`. See [Chariot MQTT Server Configuration#AccountsConfiguration](#) for details on the domains and actions available

A user with read only permissions for all services would be created by setting the permissions to ["\*.read"]



## AWS Marketplace Questions

### Where does Chariot save my sensitive information?

Below is a list of all locations on the file system where sensitive information is stored:

- `/opt/chariot/h2` - The location of the embedded H2 databases
  - Contains: Users, Roles, MQTT Credentials, License information, and Logs.
- `/opt/chariot/security` - Contains the Java KeyStore file
- `/opt/chariot/conf` - Contains Chariot configuration files
- `/opt/chariot/upload-folder` - Temporary storage for uploaded files (example: certs and keys for setting up SSL)

### How does Chariot encrypt data?

Chariot supports SSL connections to the Chariot UI and MQTT Server. Additionally Chariot encrypts sensitive data, such as passwords, using a one-way SHA-512 hashing algorithm with a configured salt.

### How do I set up and rotate SSL certificates and keys?

The following tutorials can be used for setting up and managing the Chariot server's certificates and keys:

- [Securing Chariot MQTT Server communication using SSL/TLS](#)
- [Setting up Client based authentication with Client Certificates](#)

### How do I decrypt my encrypted data?

Chariot use one-way encryption when encrypting sensitive information such as passwords and does not provide a means for decrypting the values once they are encrypted.

### How do I monitor the health of my Chariot server?

The Chariot server instance can be monitored from the Amazon EC2 console using these steps:

1. Navigate to your [Amazon EC2 console](#) and verify that you're in the correct region.
2. Choose **Instance** and select your launched instance.
3. Select the server to display your metadata page and choose the **Status checks** tab at the bottom of the page to review if your status checks passed or failed.

## Azure Marketplace Questions

### Failure to deploy with subscription error

The error indicates that the subscription you are trying to deploy into is a Microsoft CSP (Cloud Solution Provider) created subscription.

Review [Azure Marketplace offering fails to deploy with subscription error](#) for steps require to resolve

