



Configuring SSL communication (22.x.x)

September 28, 2023

Table of Contents

1 Using Waratek Keystore 1

To enable SSL communication from the Java Agent to Elasticsearch, ensure the **waratek.properties** file contains the following flag:

```
com.waratek.ElasticsearchSecure=true
```

1 Using Waratek Keystore

To use the Waratek keystore, do the following:

1. Copy the Portal Dedicated keystore, `/opt/waratek/waratek-mc-<version-build>/controller.keystore.p12` onto the agent server under the existing Waratek installation. Ideally in a location independent of the agent version or instance configuration(s) (e.g. multiple different references to `waratek.properties`).
2. Run the following command (as `root` user) in the directory in which you copied the keystore above, in order to convert the format of keystore to JKS for the agent-side.

```
# Ensuring you reference the correct executable based on your Java vendor. The
# below example assumes Jrockit as the Java vendor and not HotSpot, J9, etc.
$ <path/to>/jrockit_jdk6/jre/bin/keytool -importkeystore -srckeystore
controller.keystore.p12 -srcstoretype pkcs12 -srcalias controller -destkeystore
war_keystore.jks -deststoretype jks -deststorepass password -destalias
WaratekMCCert
```



If using Waratek Java *Upgrade* Agent, you should use the JDK version corresponding to the host JDK (and not the guest JDK) for the keytool command.

3. Edit the appropriate `waratek.properties` file(s).

```
...
com.waratek.trustStore=<absolute_directory_path_to>/war_keystore.jks
com.waratek.trustStorePassword=password # change the value for the password if
you done so in previous steps
...
```

4. Start/restart your application on Waratek and confirm it connects to the MC correctly and shows as "ONLINE" in the MC browser.