Configuring TWAdmin

Teamwork Admin (TWAdmin) is a web-based administrator tool for Teamwork Cloud. You can configure TWAdmin parameters through the Teamwork Cloud configuration file. The configuration file, **application*.conf***, is located in the directory **configuration**, where you unzipped the TWCloud install file. For example, on Linux, */opt/TeamworkCloudSuite/configuration/application.conf.*

The TWAdmin related configuration can be found under the section:

```
esi.console {
...
}
```

TWCloud will load the configuration file every time it starts. Therefore, changes made to the configuration file will be applied once you restart it. The following are the environment parameters of TWAdmin that you may want to change.

> **Warning: Changing the server protocol and server port**
>
> If you change the parameters of the server protocol and server port in the TWCloud configuration file here, you also need to change those of the authentication server's *authserver.properties* file with the same value.

Setting server protocol

TWAdmin is a web server. You can select the protocol settings between http and https by configuring the parameter "protocol". To change the setting, search for the text "protocol =" and select either **http** or **https** as the valid value. For example:

```
protocol = "http"
```

The value of the server protocol you used here must also be used for the parameters **authentication.redirect.uri.whitelist** and **twc.server.protocol** (in the authentication server's *authserver.properties* file). For example:

- **authentication.redirect.uri.whitelist**=http://<IPaddress>:8111/twcloud_admin/,https://md.redirect
- **twc.server.protocol**=http

Setting server port

The default port of TWAdmin is **8111**. You can change it by changing the value of the parameter "port". For example:

```
port = 8112
```

The value of the server port applied here must also be applied to the parameter **authentication.redirect.uri.whitelist** (in the authentication server's *authserver.properties* file).

Advance HTTPS protocol settings

By default, TWAdmin runs with an HTTPS enabled, using a self-signed certificate that is provided with the build. You can however, change the HTTPS settings.

To change the HTTPS settings

---

1. Open the TWCloud configuration file *./config/application.conf.*
2. Changing the related parameters.
3. Restart TWCloud once you have updated the server configuration.

The related configuration that you need to update when changing the HTTPS protocol settings is located in .*configuration\application.conf.*

- *esi.console. restlet.ssl. keystorePath* – to replace the path to a keystore file with the one you created, for example, "configuration/console.jks".
- *esi.console. restlet.ssl.keystorePassword* – the password to open the keystore file (if it is password-protected).
- *esi.console. restlet.ssl. keyPassword* – the password to open the private key (if it is password-protected).

Creating a Java Keystore from a CA certificate

You have to create a keystore from a CA certificate and import the CA certificate to the keystore file. If you already have the keystore file, skip this step and see the next instructions in Using an existing Keystore in TWAdmin.

To create a Java keystore, you will first need to create a .jks file that will initially only contain the private key, and then generate a CSR and generate a certificate from the CSR. Lastly, you will need to import the certificate to the keystore including any root certificates. The following commands allow you to generate a new Java Keytool keystore file, create a CSR, and import certificates. Any root or intermediate certificates will need to be imported before importing the primary certificate for your domain.

To generate a new keystore from a CA certificate

---

1. Generate a Java keystore and key pair.

   ```
   keytool -genkey -alias mydomain -keyalg RSA -keystore keystore.jks -keysize 2048
   ```

2. Generate a certificate signing request (CSR) for the Java keystore.

   ```
   keytool -certreq -alias mydomain -keystore keystore.jks -file mydomain.csr
   ```

3. Send the CSR file (*mydomain.csr*) to the authority to get a CA certificate and a signed primary certificate.
4. Import a root or intermediate CA certificate to the Java keystore.

   ```
   keytool -import -trustcacerts -alias root -file Thawte.crt -keystore keystore.jks
   ```

5. Import a signed primary certificate to the Java keystore.

   ```
   keytool -import -trustcacerts -alias mydomain -file mydomain.crt -keystore keystore.jks
   ```

Using an existing Keystore in TWAdmin

TWAdmin provides a default keystore containing a self-signed certificate (.*configration**console.jks*) after installation. If you want to use your own keystore with TWAdmin, you need to replace the default one first.

> ⚠ **TWAdmin location**
>
> - If your TWCloud version is 18.2 or 18.3
>   The location of TWAdmin should be in <*Installation directory*>*CameoEnterpriseDataWarehouse_Admin**configuration*.
> - If your TWCloud version is 18.4
>   The location of TWAdmin should be in <*Installation directory*>*TWAdmin**configuration*.

To use your own keystore with TWCloud Admin

---

1. Replace the default *console.jks* with your own keystore (the filename must be the same).
2. Update the related configuration in .*configuration**application.conf*.
3. Restart TWCloud Admin server.