

Configuring TWCloud Admin

Teamwork Cloud features the new Webapp Platform-based TWCloud Admin Console. As such, it is a standalone application that communicates with Teamwork Cloud using the REST API.

Configuration of its communication with Teamwork Cloud is located in `<install_root>/WebAppPlatform/shared/conf/webappplatform.properties`.

In this section, we will review the various settings which you may have to adjust in order to establish communications between the admin console and Teamwork Cloud. Changes to these settings are only necessary if one is not using a default installation.

```
#
# Authentication server properties
#
# Authentication server address
# http/https depending on setup of Authentication server.
authentication.server.uri=https://IP_ADDRESS:8555/authentication
```



Authserver access

If you are accessing via a hostname or FQDN, especially if you are using a signed certificate, use the applicable FQDN or hostname instead of the IP address.

```
#
# If you have configured authserver to use HTTP or to run on a different port, make sure that the URI reflects the correct values.
# Teamwork Cloud server properties
#
twc.admin.username=Administrator
twc.admin.password=Administrator
# Teamwork Cloud server address
# http/https depending on setup of Authentication server.
twc.url=https://IP_ADDRESS:8111
```



TWCloud access

Please make sure these credentials for **twc.admin.username** and **twc.admin.password** match those of a user with administrative privileges.

If you are accessing via a hostname or FQDN, especially if you are using a signed certificate, use the applicable FQDN or hostname instead of the IP address.



If you change any of the configuration parameters, you will need to restart the WebApp service.

If you have configured TWCloud to use HTTP or to run on a different port, make sure that the URI reflects the correct values.

Setting server protocol

By default, and in order to enforce a higher level of security, the admin console is accessed via HTTPS. In order to change the mode of operation to HTTP (not recommended), various configuration changes must be made.

The default port for the admin console is **8443**. In this example, we will make the changes necessary to run over HTTP on the default port of **8443**.

The WebApp server configuration is located in `<install_root>/WebAppPlatform/conf/server.xml`.

The following section:

```
<Connector executor="tomcatThreadPool"
  port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443" />
```

needs to be edited to:

```
<Connector executor="tomcatThreadPool"
  port="8443" protocol="HTTP/1.1"
  connectionTimeout="20000" />
```

The changes which we implemented consist of changing the port from **8080** to **8443**, and removing a redirect that would route to the handler on port **8443**.

Since we have configured this connector to listen on port *8443*, we now need to remove the existing connector handler on port *8443*.

The following section:

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
    sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
    maxThreads="150" SSLEnabled="true">
  <SSLHostConfig>
    <Certificate certificateKeystoreFile="../configuration/keystore.p12"
        certificateKeystorePassword="nomagic"
        type="RSA" />
  </SSLHostConfig>
</Connector>
```

needs to be commented out as follows:

```
<!--
    <Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
        sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
        maxThreads="150" SSLEnabled="true">
      <SSLHostConfig>
        <Certificate certificateKeystoreFile="../configuration/keystore.p12"
            certificateKeystorePassword="nomagic"
            type="RSA" />
      </SSLHostConfig>
    </Connector>
-->
```

By default, for security reasons, we have established a security policy requiring access to be encrypted. To disable this, we need to edit *<install_root>/WebAppPlatform/conf/web.xml*. This section is located at the very bottom of the file.

The following section:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>webapp</web-resource-name>
    <url-pattern>/*</url-pattern>
  </web-resource-collection>
  <user-data-constraint>
    <transport-guarantee>CONFIDENTIAL</transport-guarantee>
  </user-data-constraint>
</security-constraint>
```

needs to be edited as follows:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>webapp</web-resource-name>
    <url-pattern>/*</url-pattern>
  </web-resource-collection>
  <user-data-constraint>
    <transport-guarantee>NONE</transport-guarantee>
  </user-data-constraint>
</security-constraint>
```

In the next example, we will configure the Admin Console to run HTTPS on a different port (*8444*).


The following code section:

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
    sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
    maxThreads="150" SSLEnabled="true">
  <SSLHostConfig>
    <Certificate certificateKeystoreFile="../../configuration/keystore.p12"
      certificateKeystorePassword="nomagic"
      type="RSA" />
  </SSLHostConfig>
</Connector>
```

needs to be edited as follows:

```
<Connector port="8444" protocol="org.apache.coyote.http11.Http11NioProtocol"
    sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
    maxThreads="150" SSLEnabled="true">
  <SSLHostConfig>
    <Certificate certificateKeystoreFile="../../configuration/keystore.p12"
      certificateKeystorePassword="nomagic"
      type="RSA" />
  </SSLHostConfig>
</Connector>
```

As can be seen, the only change is the definition of the port number, which changed from 8443 to 8444.

 If you change either the protocol or the port from the default, you need to edit **authentication.redirect.uri.whitelist**, located in `<install_root>/AuthServer/config/authserver.properties` accordingly.

Related pages:

- [Accessing TWCloud Admin](#)
- [Disabling authentication with user name and password](#)