Application Note – CommanderConnect

Integration of an SSL certificate

The SSL certificate, which is integrated into the Commander Connect software, should be an applied for certificate and not a self-signed one. Self-signed certificates are blocked by web browsers because they cannot be confirmed. This would require an exception to be defined in the web browser, at least for the Commander. The warning must therefore be confirmed.

If you want to apply for a certificate, you need a unique server name (FQDN), because no certificates can be issued for "localhost" or an internal IP address. The name itself is stored in the certificate and is checked during the application and later also compared in the browser. For example, if the certificate has been issued for "deister.com", but "deister.de" has been called up, the web browser will display a warning. This does not necessarily mean that the Commander must be permanently accessible from outside (Internet). A connection to the company's own DNS server is sufficient, which redirects the server name to the internal IP address. Here the web browser only checks the web page called up and compares this name with the name stored in the certificate.

Replacing the SSL certificate in the Server.xml file of Commander 4.8

The exchange of a new SSL certificate takes place in the file "Server.xml" in the area "Connector protocol" ("...\Commander 4\Basic\WebUserInterface\conf").

SSLPasswordD: The password for the certificate issued during the application (using the example password "deisterTest0815")
SSLCertificateFile: The certificate (in this example file name "cert.pem")
SSLCertificateKeyFile: The key (in this example file name "key.pem")

```
<Connector protocol="org.apache.coyote.http11.Http11AprProtocol" port="8443" server="Apache"
        URIEncoding="UTF-8" disableUploadTimeout="true"
        acceptCount="100" enableLookups="false" minSpareThreads="25"
        maxThreads="200" maxHttpHeaderSize="8192" SSLPassword="deisterTest0815" SSLCertificateFile="cert.pem"
        SSLCertificateKeyFile="key.pem"
        SSLProtocol="TLSv1.2" SSLEnabled="true" secure="true" scheme="https"
        SSLCipherSuite="ALL:!aNULL:!eNULL:!SSLv2:!LOW:!3DES:!SEED:!ECDSA:!RC4:!EXP:!IDEA"
        SSLDisableCompression="true" SSLHonorCipherOrder="true"/>
```

A self-signed SSL certificate from deister electronic is supplied ("...\Commander 4\Basic\WebUserInterface"). The easiest way is to exchange the files, change the password in the file "Server.xml" and restart the ApacheTomcat service. Via the request https://<Servername>:8443 the SSL encryption can be used.

If the customer wants to use a different port, this must also be adjusted in the file "Server.xml" ("443" is the default and does not have to be additionally specified in the browser address bar).

File "Server.xml" from Commander 4.9

As of Commander 4.9, the Server.xml file has a different structure.

certificateKeyPassword: The password for the certificate issued during the application (usingthe example password "deisterTest0815")
certificateFile: The certificate (in this example file name "cert.pem")
certificateKeyFile: The key (in this example file name "key.pem")

```
<SSLHostConfig
ciphers="ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-
SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384"
disableSessionTickets="true"
honorCipherOrder="false"
protocols="TLSv1.2,TLSv1.3">

<Certificate
certificateFile="cert.pem"
certificateKeyFile="key.pem"
certificateKeyPassword="deister0815" />
</SSLHostConfig>
```

Switching to an encrypted communication

If the use of "http://..." is no longer possible and the user should switch to an encrypted communication "https://...", this can be adjusted in the file "Web.xml". Alternatively, port 8095 is no longer released in the firewall. There are two possibilities for the implementation:

a. Only the Commander redirects automatically:

„..\Commander 4\Basic\WebUserInterface\webapps\ROOT\WEB-INF“ (File: „Web.xml”)

To do this, reinsert the prepared block (is stored as a comment):

```
<!-- <user-data-constraint>
<transport-guarantee>CONFIDENTIAL</transport-guarantee>
</user-data-constraint>-->
```

b. All websites are redirected globally:

„..\Commander 4\Basic\WebUserInterface\conf“ (File: „Web.xml”)

Since all "servlets" commands must be loaded before the change, the block below must be inserted after all <servlet> commands.

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