



Module „Secure-Connect“

Manual for installation
and usage of the module
“Secure-Connect”



Table of Contents

| | |
|---|----|
| 1)Contents of the package..... | 3 |
| 2)Features of the module..... | 4 |
| 3)Installation of the module..... | 5 |
| Step 1: Installation of the module-file..... | 5 |
| Step 2: Generate the certificates and key-stores..... | 5 |
| Step 3: Preparing the MySQL-Server for SSL-functionality..... | 8 |
| Step 4: Preparing Labmatica LIMS for SSL-functionality..... | 9 |
| Step 5: Create a directory for storing the user-data..... | 9 |
| Step 6: Edit the Labmatica-configuration file..... | 9 |
| Step 7: First start of the system..... | 10 |
| Step 8: Managing other users of Labmatica LIMS..... | 11 |
| Step 9: Every following start of the system..... | 11 |
| Installation Remark..... | 12 |
| 4)Messages of the module..... | 13 |
| 5)Configuration of the module..... | 15 |



1) Contents of the package

The package consists of a Zip-file containing the following files:

- 1 labmatica_secure_connect.jar The module "Labmatica-Secure-Connect"
- 2 create_cert.bat The batch-file to generate the certificates on Windows
- 3 create_cert.sh The shell-script to generate the certificates on Linux
- 4 Secure-Connect-Manual.pdf A copy of this guide as PDF

2) Features of the module

The features of the module "Labmatica-Secure-Connect" are the following:

- Protection of the database-connection-data against unauthorized access
- Securing the transfer of data through SSL-encryption
- Changing the database-connection-data in Labmatica LIMS
- Protection against unauthorized removal of the module
- Limitation to maximum 3 attempts of login during a system-start
- Demand the user to change his password after 365 days



3) Installation of the module

Step 1: Installation of the module-file

To install the module “Labmatica-Secure-Connect“, you first have to perform the following steps:

1. Unzip the file that was shipped with the package
2. Copy the file “labmatica_secure_connect.jar” into the Lib-folder of your Labmatica-installation

Step 2: Generate the certificates and key-stores

Now, you have to generate the certificate-authority (CA), the certificates and the key-stores. This is necessary to enable the SSL-ability between the MySQL-Server and Labmatica LIMS. So, first of all, you have to install the Software “OpenSSL”. A version according to your operating system can be found at and downloaded from <http://www.openssl.org>.

Note: You may have to set an environment-variable to the Bin-folder of your OpenSSL-installation, in order to be able to do next steps.

Note: The SSL-functionality only works with MySQL 4.0.4 or higher. You may also require JDK 1.4.1 or newer.

Once you have installed OpenSSL, you can continue to create the certificate authority by running the batch-file “create_cert.bat” on Windows or the shell-script “create_cert.sh” on Linux. To do so, open a prompt/terminal, change into the appropriate directory and type in:

- On Windows: create_cert.bat
- On Linux: ./create_cert.sh

Note: On Linux your may have to set the permission to execute the file with the command:

```
chmod 777 create_cert.sh
```

1. After the first 3 automatic steps, which do some preparations, in step 4 of the script, the previously created Certificate Authority’s private key will be self signed by the administrator. The data entered should accord to your situation and may look like this:

```

micgan@ToshibaLinux:~/SSL - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
*****
* Step 4: Self-sign CA's Public Key Certificate *
*****
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [DE]:DE
State or Province Name (full name) [Hessen]:Hessen
Locality Name (eg, city) []:Frankfurt
Organization Name (eg, company) []:Laboptimizer Ltd.
Organizational Unit Name (eg, section) [Labmatica LIMS]:Labmatica LIMS
Common Name (eg, YOUR name) []:John Doe
Email Address []:john.doe@email-provider.com

```

Note: The Common Name must be different in all following information requests of this form.

2. The following step 5 is also an automatic step. It automatically creates the server-key. After that, you have to create the server's signing request in step 6. Here, remember, that the common name has to be different to the name previously entered. In step 7, the request is then signed by the local CA:

```

micgan@ToshibaLinux:~/SSL - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
*****
* Step 5: Generate Server-Key *
*****
Generating RSA private key, 1024 bit long modulus
.....+++++
.....+++++
e is 65537 (0x10001)

*****
* Step 6: Create Server's Certificate Signing Request *
*****
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [DE]:DE
State or Province Name (full name) [Hessen]:Hessen
Locality Name (eg, city) []:Frankfurt
Organization Name (eg, company) []:Laboptimizer Ltd.
Organizational Unit Name (eg, section) [Labmatica LIMS]:Labmatica LIMS
Common Name (eg, YOUR name) []:Server side
Email Address []:.

*****
* Step 7: Sign Server-Request CA-Certificate *
*****
Signature ok
subject=/C=DE/ST=Hessen/L=Frankfurt/O=Laboptimizer Ltd./OU=Labmatica LIMS/CN=Server side
Getting CA Private Key

```



Note: Alternatively, you can use a SSL-provider, just like Verisign, to sign the server's signing request.

3. In step 8, the client key will be created. Note that the first entered information corresponds to the common name and must be different to all previously entered names. At the end of this step, you will be asked, if you trust the certificate. Here, please type in the word for "yes", according to your language:

- English: "Yes"
- Deutsch: "Ja"
- Français: "Oui"

After step 8, in steps 9 and 10, the client's signing request is created and automatically signed by the local CA.

```
micgan@ToshibaLinux:~/SSL - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
*****
* Step 8: Generate Client's Private Key *
*****
Wie lautet Ihr Vor- und Nachname?
[Unknown]: Client Side
Wie lautet der Name Ihrer organisatorischen Einheit?
[Unknown]: Labmatica LIMS
Wie lautet der Name Ihrer Organisation?
[Unknown]: Laboptimizer Ltd.
Wie lautet der Name Ihrer Stadt oder Gemeinde?
[Unknown]: Frankfurt
Wie lautet der Name Ihres Bundeslandes oder Ihrer Provinz?
[Unknown]: Hessen
Wie lautet der Landescode (zwei Buchstaben) für diese Einheit?
[Unknown]: DE
Ist CN=Client Side, OU=Labmatica LIMS, O=Laboptimizer Ltd., L=Frankfurt, ST=Hessen, C=DE richtig?
[Nein]: Ja

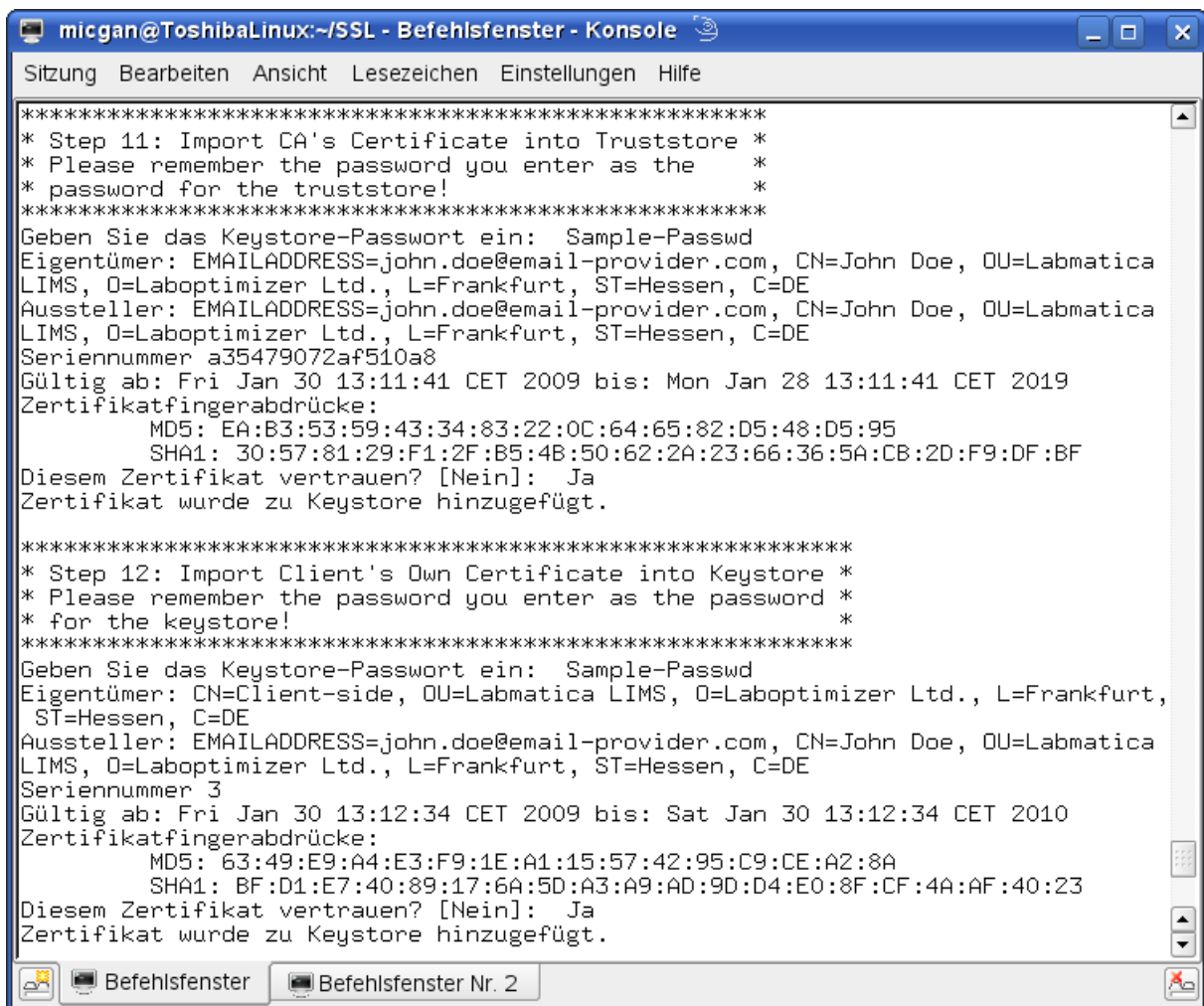
*****
* Step 9: Generate Client's CSR *
*****

*****
* Step 10: Sign Client's CSR *
*****
Signature ok
subject=/C=DE/ST=Hessen/L=Frankfurt/O=Laboptimizer Ltd./OU=Labmatica LIMS/CN=Client Side
Getting CA Private Key
```

Note: Again, you can use a SSL-provider, just like Verisign, to sign the client's signing request.

4. In the following steps 11 and 12, the certificates created are imported in the keystores for the client side. Here, you first have to enter a password for the truststore and the keystore. It is important that you remember these two passwords! Then you have to confirm that you trust the certificates with the word for “yes”, according to your language:

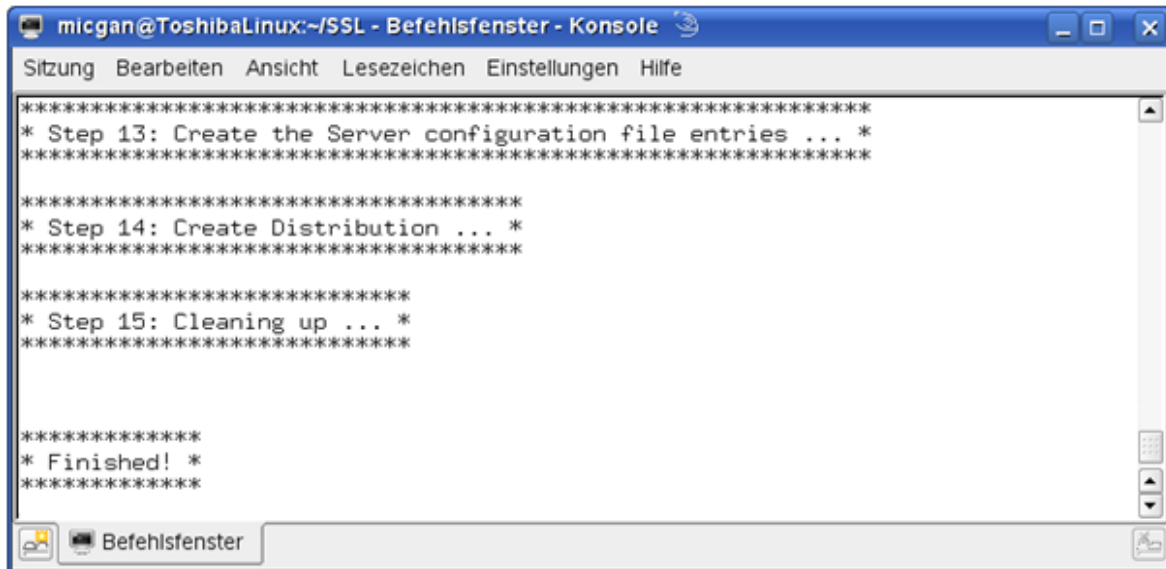
- English: “Yes”
- Deutsch: “Ja”
- Français: “Oui”



```
micgan@ToshibaLinux:~/SSL - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
*****
* Step 11: Import CA's Certificate into Truststore *
* Please remember the password you enter as the *
* password for the truststore! *
*****
Geben Sie das Keystore-Passwort ein: Sample-Passwd
Eigentümer: EMAILADDRESS=john.doe@email-provider.com, CN=John Doe, OU=Labmatica
LIMS, O=Laboptimizer Ltd., L=Frankfurt, ST=Hessen, C=DE
Aussteller: EMAILADDRESS=john.doe@email-provider.com, CN=John Doe, OU=Labmatica
LIMS, O=Laboptimizer Ltd., L=Frankfurt, ST=Hessen, C=DE
Seriennummer a35479072af510a8
Gültig ab: Fri Jan 30 13:11:41 CET 2009 bis: Mon Jan 28 13:11:41 CET 2019
Zertifikatfingerabdrücke:
    MD5: EA:B3:53:59:43:34:83:22:0C:64:65:82:D5:48:D5:95
    SHA1: 30:57:81:29:F1:2F:B5:4B:50:62:2A:23:66:36:5A:CB:2D:F9:DF:BF
Diesem Zertifikat vertrauen? [Nein]: Ja
Zertifikat wurde zu Keystore hinzugefügt.

*****
* Step 12: Import Client's Own Certificate into Keystore *
* Please remember the password you enter as the password *
* for the keystore! *
*****
Geben Sie das Keystore-Passwort ein: Sample-Passwd
Eigentümer: CN=Client-side, OU=Labmatica LIMS, O=Laboptimizer Ltd., L=Frankfurt,
ST=Hessen, C=DE
Aussteller: EMAILADDRESS=john.doe@email-provider.com, CN=John Doe, OU=Labmatica
LIMS, O=Laboptimizer Ltd., L=Frankfurt, ST=Hessen, C=DE
Seriennummer 3
Gültig ab: Fri Jan 30 13:12:34 CET 2009 bis: Sat Jan 30 13:12:34 CET 2010
Zertifikatfingerabdrücke:
    MD5: 63:49:E9:A4:E3:F9:1E:A1:15:57:42:95:C9:CE:A2:8A
    SHA1: BF:D1:E7:40:89:17:6A:5D:A3:A9:AD:9D:D4:E0:8F:CF:4A:AF:40:23
Diesem Zertifikat vertrauen? [Nein]: Ja
Zertifikat wurde zu Keystore hinzugefügt.
```

5. The last steps 13 to 15 are automatic steps again and finish up the setup of the SSL-certificates:



```
micgan@ToshibaLinux:~/SSL - Befehlsfenster - Konsole
Sitzung Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
*****
* Step 13: Create the Server configuration file entries ... *
*****

*****
* Step 14: Create Distribution ... *
*****

*****
* Step 15: Cleaning up ... *
*****

*****
* Finished! *
*****
```

6. After these steps, there are two folders in the working directory, `server_files` and `client_files`.

The `server_files`-folder contains the following files:

- `ca-cert.pem`
- `server-cert.pem`
- `server-key.pem`
- `my.txt`

The `client_files`-folder contains the following files:

- `truststore`
- `keystore`

Step 3: Preparing the MySQL-Server for SSL-functionality

To prepare the MySQL-Server for the SSL-functionality, do the following steps:

- On the MySQL-Server-machine, create a new directory
- Copy all the files of the `server_files`-folder into this directory
- Now, you have to configure the server for SSL-functionality:



1. Open the created configuration-files-entries file, my.txt in the server_files-folder and copy all its contents
2. Open the configuration-file of the MySQL-server in an editor and navigate to the section called [mysqld]. The configuration-files is located at:
 - /etc/my.cnf on Linux
 - my.ini in the Installation-path on Windows

Now paste the copied contents into a free space in this section.

- Start/Restart the server in the manner you did it so far

Step 4: Preparing Labmatica LIMS for SSL-functionality

To prepare Labmatica LIMS for SSL-functionality, you have to do the following:

- In the Labmatica-folder, create a new directory
- Copy the files “.truststore” and “.keystore” of the client_files-folder into this directory

Step 5: Create a directory for storing the user-data

Now it's time to create a directory for the later storing of the encrypted user-files. These files are used to store the encrypted database-connection-data. For example, this directory can be created in the installation-folder of Labmatica LIMS and should also be protected by appropriate OS-privileges.

Step 6: Edit the Labmatica-configuration file

In this step, you have to edit the configuration-file of Labmatica LIMS. For now, this file is called “connect.xml” and should be located in the root-directory of the Labmatica-installation. So, to edit the file, please do the following:

1. Open the file with an editor
2. Edit the following tags:

```
<userid>...</userid>
```

```
<password>...</password>
```

Replace the containing values by “#####”

3. Edit the following tags:

```
<!-- Where should the keystore- and user-files be stored? -->
```

```
<userdirectory>XXXX</userdirectory>
```

Please set here the path to the directory, where the user-files should be stored. This is the directory created in step3.

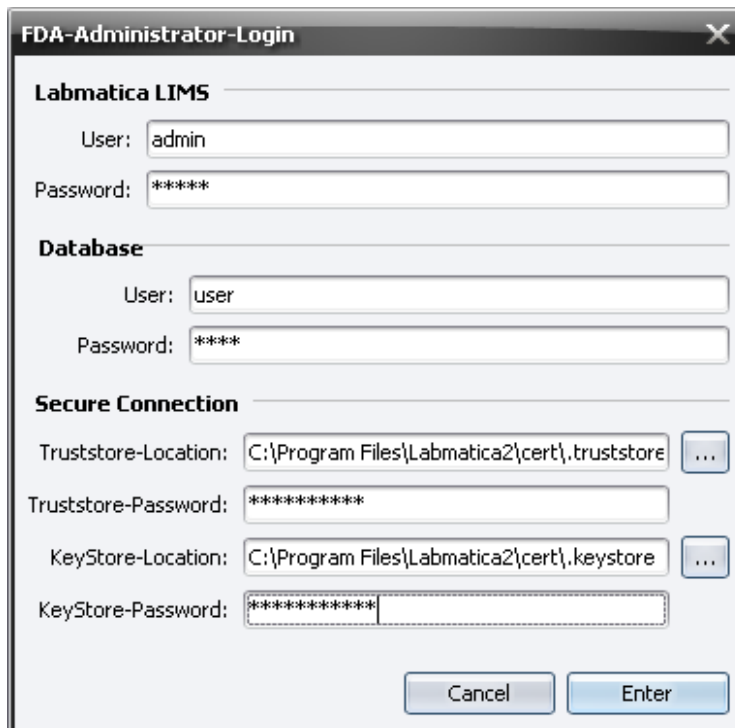
<!-- What is the default-language for the Secure-Connect-Module? English, Francais or Deutsch -->

<default_language>XXXX</default_language>

Please set here your preferred language for the Secure-Connect-Module. This will be the language, used in the login masks.

Step 7: First start of the system

Now, the system is ready to use the module and you can start the system. When you do so, after the start-progress the following window will appear:



This is the login-window for the administrator, which is used to initialize the Secure-Connect-Module. Here, please enter the following information:

- Labmatica LIMS:
 - o User: YOUR username or “admin”

- Password: YOUR password or “admin”
- Database:
 - User: The database-user
 - Password: The password of this user
- Secure-Connection:
 - Truststore-Location: The path of the trust-store file “.truststore” previously created
 - Truststore-Password: The password of the trust-store previously created
 - Keystore-Location: The path of the key-store file “.keystore” previously created
 - Keystore-Password: The password of the key-store previously created

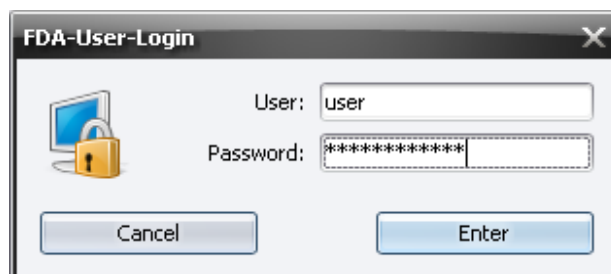
Step 8: Managing other users of Labmatica LIMS

After logging in as administrator, you have to create the files for all other users. This can be made by the following procedure:

1. Call the Users-tab of Labmatica LIMS
2. For every user do the following:
 - a. Assign a new password
 - b. Save the settings
3. Let the user know the new password. The next time the user starts the system he has to change it

Step 9: Every following start of the system

Now, the system is completely initialized and all needed data has been created. Each user can now use the system as usual. That means that from now on the login-window will look as follows:



Installation Remark

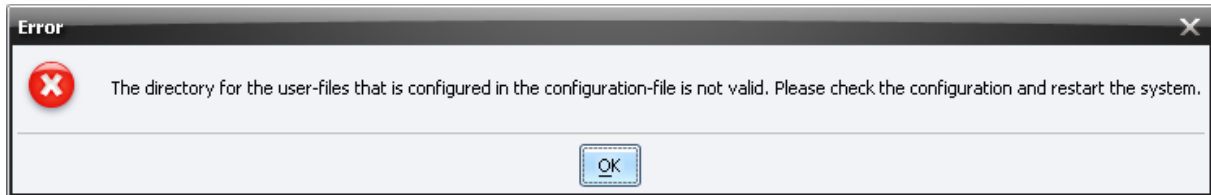
Removing the module from the Lib-folder after installation will cause an inability of starting the system. To uninstall the module you will first have to uncheck the specified checkbox in

the Modules-panel of the configuration and save the settings. Only then you will be able to remove the module and continue working with the system.

4) Messages of the module

After installation, the functionality of the module runs completely in background. However, there may come up some messages, when the proper functionality cannot be assured.

Here are the explanations of these messages and their possible solution:



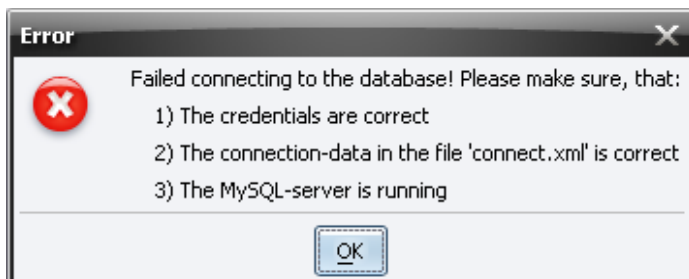
Explanation:

This error-message occurs during the start of the system. The reason is that the value of the tag `<userdirectory>` in the configuration file `connect.xml` is either not valid or references a directory that doesn't exist.

Solution:

To solve this problem try the following:

1. Open the file "connect.xml" in an editor
2. Check the value of the tag `<userdirectory>`



Explanation:

This error-message occurs during the first administrator-login from step 7. The Appearance

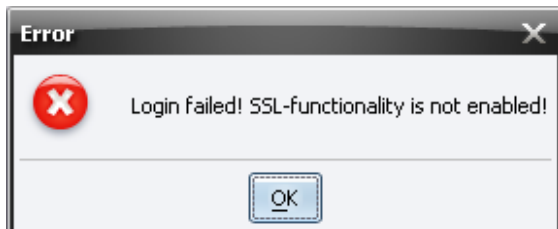
can have three reasons:

1. The entered credentials for the database-account are incorrect
2. The additional database-connection-data in the configuration-file “connect.xml” is incorrect
3. The database-server is not running or not reachable

Solution:

To solve this problem try the following:

1. Check and re-enter the account-data for the database
2. Open the file “connect.xml” in an editor and check the tags “<dbasetype>” and “<URL>” for correct values
3. Make sure, that the database-server is running and reachable:
 - a. You can test this by trying to connect to the database with the MySQL-Query-Browser
 - b. Make sure that the firewall is properly configured

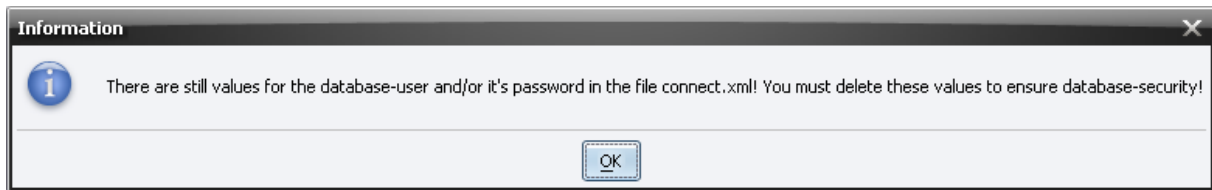


Explanation:

This error-message appears in both, the first administrator login and every following login, in the case that the database-server is running, but the SSL-functionality is not enabled.

Solution:

To solve this problem, you have to start the MySQL-Server with the proper command-line options. You can use the batch-file “start_mysql_w_SSL.bat” for this.



Explanation:

This message appears, every time an administrator logs into the system in the case, that there is still database-account-data in the configuration-file "connect.xml".

Solution:

To solve this problem, do the following:

1. Open the file "connect.xml" in an editor
2. Set the values of the following tags to #####

`<userid>XXXX</userid>`

`<password>XXXX</password>`

5) Configuration of the module

The configuration-tab of this module is reachable over the main-configuration-tab of the system in the main-window. There is a tab called “FDA-Secure-Connect”. As you can see in the picture below, here you can change the database-connection-data. This can be used in the case, that the currently used account has to be changed or that new certificates had to be created. When you change the account-data and save the configuration by the button in the upper left, then every users file will be changed, in order to be available at the next start of the system.

