License Asset Manager™

Application Note

LAMUM Windows HTTPS/SSL/TLS Conversions

This document presents recommended steps to convert **LAMUM** software to operate in a secure environment. These chiefly entail reconfiguration to the two web servers (Apache and Apache Tomcat) which support the **LAM** and **UM** components, such that they "speak" to the end user's browser over Industry standard secure (encrypted) data transfer protocol(s) best known by the acronyms **HTTPS** (secure http), **SSL** (Secure Sockets Layer), and **TLS** (Transport Layer Security), all of which refer to the same technology. We'll use **TLS** to refer to this in this document where necessary.

It includes a listing of configuration and application files and their locations, recommended changes to each, and reference to two key web resources which we used to guide our conversion effort.

Approach taken.

Both Apache web server and Apache Tomcat (hereafter referred to simply as Tomcat) JSP/servlet container support serving web content over secure connections. We found very useful web links to instructions for making the necessary Apache changes (http://rubayathasan.com/tutorial/apache-ssl-on-windows/) (hereafter called the "Apache page document"), and Tomcat changes (https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html) (hereafter called the "Tomcat page document"). Since we will refer to these often in the following discussion, we recommend readers have ready access to each, either in online or printed form, or both.

Configuration changes: Apache.

1) Program file to be replaced. Latest version of the cURL utility program used by LAM/UM. Supports TLS (and standard HTTP).

curl.exe (version 7.57.0). Binary download link:
https://dl.uxnr.de/build/curl/curl_winssl_msys2_mingw64_stc/curl-7.57.0/curl-7.57.0.zip

Open zip file and navigate to src directory; **curl.exe** is there. Rename older files by appending an 'x' to the end of the file extension, then copy new curl.exe to <LAMUM-path>\curl\. Once testing is completed successfully, you may remove these files.

Test this by going to **Admin** tab and clicking on **Run Cron** button. Expect file <LAMUM-path>\cron\etc\taskinfo.txt timestamp to be updated. This shows that the new **curl.exe** still supports standard HTTP, since we haven't changed any configurations yet.

At this point, since we are about to generate and add TLS certificate and key files, a JKS (Java KeyStore) keystore, followed by changing configurations of both Apache and Tomcat, we recommend shutting down the Tomcat, Apache, MariaDB, and Cron services, then backing up the complete LAMUM folder before proceeding.

*** Please refer to Apache page document at: http://rubayathasan.com/tutorial/apache-ssl-on-windows/ for the remainder of this section.

2) New files to add. Self-signed certificate and private key files.

(Please use the input and output file names supplied in the boldface commands for simplicity.)

In a command line window (run as administrator), navigate to <LAMUM-path>\apache\bin. Refer to the Apache page document, section titled "Creating a self-signed SSL Certificate using OpenSSL," and follow these instructions.

To create the SSL certificate we will need the openssl.cnf files location but the default location set by OpenSSL for this file is setup according to a Linux distribution, so we need to fix it for Windows.

We need to **setup the Windows environment variable** OPENSSL_CONF to point to the openssl.cnf file's location. It is located in "<LAMUM-path>\apache\conf\" directory.

We can set it up by the following command or through the GUI interface:

```
set OPENSSL_CONF=<LAMUM-path>\apache\conf\conf\openssl.cnf
```

All files generated from the following commands will reside in "<LAMUM-path>\apache\bin" folder.

Now that we have the environment variable set we need to create a new OpenSSL certificate request using the following command:

```
openssl req -new -out server.csr
```

It will ask you some questions and you can safely ignore them and just answer the following questions:

PEM pass phrase: Password associated with the private key you're generating (anything of your choice).

Common Name: The fully-qualified domain name associated with this certificate (*i.e.* www.your-domain.com).

Now we need to remove the passphrase from the private key. The file "server.key" created from the following command should be only readable by the apache server and the administrator. You should also delete the .rnd file because it contains the entropy information for creating the key and could be used for cryptographic attacks against your private key.

```
openssl rsa -in privkey.pem -out server.key
```

Now we need to set up an expiry date, it could be any time of your choice, we use 365 days below:

```
openssl x509 -in server.csr -out server.cert -req -signkey server.key -
days 365
```

We have the Self-signed SSL certificates ready now. Now we need to MOVE the "server.cert" and "server.key" files to the "<LAMUM-path>\apache\conf" location.

3) Existing files to be modified, following instructions in same Apache page document, now in section titled "Configuring Apache to run SSL/HTTPS server."

Now that we have the Self-signed SSL certificate ready, all we need is to configureApache to start the SSL server.

First we modify the "<LAMUM-path>\apache\conf**httpd.conf**" file.

Open up <LAMUM-path>\apache**httpd.conf** in a text editor and look for the lines:

LoadModule ssl_module modules/mod_ssl.so and remove any pound sign (#) characters preceding it.
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so and remove any pound sign (#) characters preceding it.
Include conf/extra/httpd-ssl.conf and remove any pound sign (#) characters preceding it.

Comment out [with (#) sign] any active Listen statements. These will be replaced by corresponding ones in <LAMUM-path>\apache\conf\extra**httpd-ssl.conf**.

Modify <LAMUM-path>\apache\conf\extra**httpd-ssl.conf** thus:

Replace “Listen 443” line with the “Listen” (without quotes) lines commented out in <LAMUM-path>\apache\conf**httpd.conf**, ensuring that any comments are removed (especially after a cut and paste). Replace all other occurrences of port number “443” with “8182” (without quotes).

Correct folder prefix in line **SSLSessionCache** with proper <LAMUM-path>/apache/ prefix. (Note the forward slashes here are intended to avoid misinterpretation of backslashes.)

Correct VirtualHost entries thus:

Correct folder prefixes for **DocumentRoot**, **ErrorLog**, and **TransferLog** keywords to read as your local <LAMUM-path>/apache/ prefix (again, forward slashes here are intended).

Correct **ServerName** parameter to match the “Listen” *yourIP_addr*:**8182** entry changed earlier.

Supply proper email address of LAM/UM server administrator, *e.g.* “admin@yourco.com” (without quotes).

Correct folder prefix for **SSLCertificateFile** to read as your local <LAMUM-path>/apache/ prefix (again, forward slashes here are intended). Correct “server.crt” to read “server.c**e**rt” (without quotes).

Correct folder prefix for **SSLCertificateKeyFile** to read as your local <LAMUM-path>/apache/ prefix (again, forward slashes here are intended). Here “server.key” is already correct.

Correct folder prefix for CustomLog to read as your local <LAMUM-path>/apache/ prefix (again, forward slashes here are intended).

FOR EXISTING CRON JOBS

Modify the following files, changing the "**http**" fields to "**https**" (without quotes):
<LAMUM-path>\cron\cron.tab

```
Change all entries with http to https

# Entry for backing up database (days)
0 0 */7 * * curl.exe
http://192.168.1.110:8182/lammonitor/config/makeBackup

# Entry for long checkout notification
3 * * * *  curl.exe http://192.168.1.110:8182/lammonitor/alert

# Entry for backing up database (days)
0 0 */7 * * curl.exe
https://192.168.1.110:8182/lammonitor/config/makeBackup

# Entry for long checkout notification
3 * * * *  curl.exe https://192.168.1.110:8182/lammonitor/alert
```

Modify the following files as shown below:
<LAMUM-path>\apache\htdocs\lammonitor\conf\config.php (starting with line 4):

from:

```
$config['base_url'] = 'http://yourIP_addr:8182/lammonitor/';
```

to:

```
$config['base_url'] = `https://yourIP_addr:8182/lammonitor/';
```

Configuration changes: Tomcat.

Please refer to Tomcat page document at: https://tomcat.apache.org/tomcat-8.0-doc/ssl-howto.html for this section.

1) New file to add. Self-signed certificate (**.keystore)** file.

In a command line window (run as administrator), navigate to your "home" folder. This is where the Java program **keytool** will put its output file (**.keystore**) by default. Refer to the Tomcat page document, section titled "Configuration," subtitled "Prepare the Certificate Keystore" and follow these instructions for Windows.

To create a new `JKS` keystore from scratch, containing a single self-signed Certificate, execute the following from a terminal command line:

Windows:

```
"%JAVA_HOME%\bin\keytool" -genkey -alias tomcat -keyalg RSA
$JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA
```

(The RSA algorithm should be preferred as a secure algorithm, and this also ensures general compatibility with other servers and components.)

This command will create a new file, in the home directory of the user under which you run it, named "**.keystore**". To specify a different location or filename, add the –keystore, followed by the complete pathname to your keystore file, to the **keytool** command shown above. You will also need to reflect this new location in the **server.xml** configuration file, as described later. For example:

Windows:

```
"%JAVA_HOME%\bin\keytool" -genkey -alias tomcat -keyalg RSA
    -keystore \path\to\my\keystore
```

In most cases, **keytool** will prompt to convert the key just generated (in **.keystore**) to PKCS12 format. Because the PKCS12 format is an internet standard, we recommend doing this conversion, as other certificate tools, among them OpenSSL, can also handle PKCS12 certificates. The exact command will display as the last lines of **keytool** output. In case **keytool** does not issue this prompt, use the following (we recommend placing these two into command ["batch"] files); the first performs the conversion, the second tests and comfirms the conversion:

(cvtjks2pkcs12.cmd):

```
keytool -importkeystore -srckeystore .keystore -destkeystore .keystore -
srcstoretype JKS -deststoretype PKCS12 -deststorepass changeit
```

(chkjks2pkcs12.cmd):

```
keytool -list -V -keystore .keystore -storetype pkcs12
```

Please use the input and output file names supplied in the examples for simplicity, and preserve the Tomcat default keystore password of "**changeit**" to simplify things, as described in the instructions. After executing the command, move the file "**.keystore**" to your local <LAMUM-path>\apache-tomcat\conf.

2) Existing files modified, with brief description of change.

server.xml: addition of new TLS-enabled Connector element to Service element, replacing the Connector designed for standard HTTP, which is to be commented out. This specifies parameters for a TLS-enabled connection to Tomcat, in folder <LAMUM-path>/apache-tomcat/conf/.

Add a new TLS Connector element under the existing commented-out entry, leaving the commented-out as it is, using the following entry setting up port **8181** as Tomcat's TLS port (Note the forward slashes in the file path here are intended to avoid misinterpretation of backslashes.):

```
<Connector port="8181"
      protocol="org.apache.coyote.http11.Http11Nio2Protocol"
      maxThreads="200"
      SSLEnabled="true"
      scheme="https"
```

```
secure="true"
keystoreFile="C:/LAMUM/apache-tomcat/conf/.keystore"
keystorePass="changeit"
clientAuth="false"
sslProtocol="TLS" />
```

Modify the following Connector, changing redirect port (from 8443, usually) to 8181:

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8181" />
```

Modify the following files, changing the "**http**" fields to "**https**" (without quotes):
<LAMUM-path>\apache-tomcat\webapps\lam\lamview\output**DaemonManager.jsp**
<LAMUM-path>\apache-tomcat\webapps\lam\lamview\output**Usage.jsp**

Restart and Test.

At this point, it's time to test the changes. Start the cron, MariaDB, Tomcat, and Apache services using the Services snap-in dialog from the Computer Management application. For each service, wait until the "Restart the service" line appears, before moving to the next. Since we haven't re-configured cron or MariaDB, it is unlikely that there should be any issues with these, but wait, just to be certain these startups are OK.

1) Restart services and check startup status.
Starting Tomcat and Apache services may encounter issues, where Windows will put up a dialog indicating a problem, which is most likely to be a missed step or an error in re-configuration. Looking at Event Viewer (also under Computer Management), click System, will show event messages, and Error events here pertaining to service startup are the best source of pinpointing what may have caused the startup failure. Look back to the appropriate "Configuration Changes: ..." section to find the step where there was either an error or omission. Make the necessary change, save the file involved, and attempt to start the service again. Once you see the "Restart the service" line, it is safe to move on to start the next service. If errors occur on this service start, repeat the steps described in this paragraph until the service starts properly. (There is no required sequence/order for starting either the Tomcat or the Apache service).

2) Run LAM/UM. Look for any issues and resolve.

Start LAM/UM, logging in. Remember to use the secure "scheme" name "**https**" instead of "**http**" (without quotes) to get to the login screen. If not you will see a "400 Bad Request" (or similar) message (meaning you forgot to use "**https**" to start).

A simple way to check that LAM/UM is running as expected under the new secure connection is to start by logging in to:

https://YourIPaddress:8181/lam/

Once logged in, expect to see the initial (Expirations) tab/page. If not, there are likely configuration change errors to be corrected. Otherwise click on each successive tab, "PO",

"Licenses", "Usage", "Vendors", "Tools", "Servers", "Reports", "Settings", and "Daemon", checking for each page coming up as expected. If the "Daemon" tab fails to display, there is probably a missed change to <LAMUM-path>\apache-tomcat\webapps\lam\lamview\output**DaemonManager .jsp**, where "**`http`**" has not been converted to "**`https`**".

Now go back to the "Usage" tab, and check that the "Home" tab/page displays the "Licenses being monitored" title and data table. If not, there is probably a missed change to <LAMUM-path>\apache-tomcat\webapps\lam\lamview\output**Usage.jsp**, where "**`http`**" has not been converted to "**`https`**". Otherwise, continue across the remaining tabs under "Usage": "Current", "History", "Denials", "Users", "Groups", "Batch", "Alerts", "Parsers", and "Admin", checking for each page coming up as expected.

3) Special considerations for supported browsers and the Usage and Daemon tabs.

If either the Usage or Daemon tab fails to display despite having changed **Usage.jsp** or **DaemonManager .jsp** as described above, instead showing a message and graphic indicating the browser could not connect to the web address expected, do the following in a new browser tab. In the address bar, enter the actual URL as found in each JSP file as follows:

https://yourIP_addr:8182/lammonitor/ (for Usage tab)

and

https://yourIP_addr:8182/daemonmanager/ (for Daemon tab)

3a – Chrome [tested with version 63]).

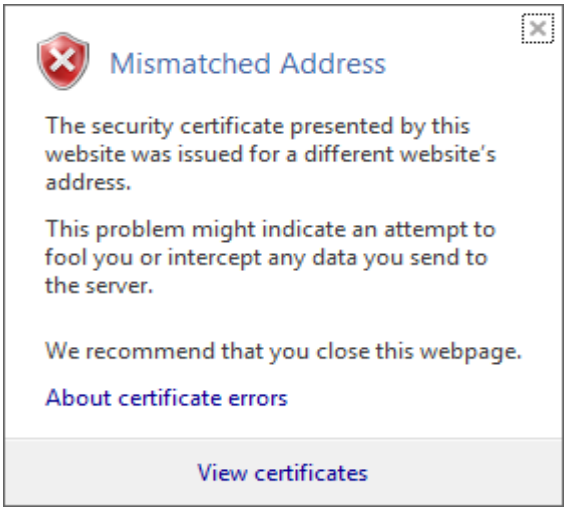
In each case, Chrome will respond with a warning page about an insecure (self-signed) certificate on the web server. Bypass this warning and accept (click) the option to load the page anyway. You should now be able to go back to the LAM/UM Usage or Daemon tab, refresh that page, and the proper information should now be displayed.

3b – Internet Explorer [tested with version 11])

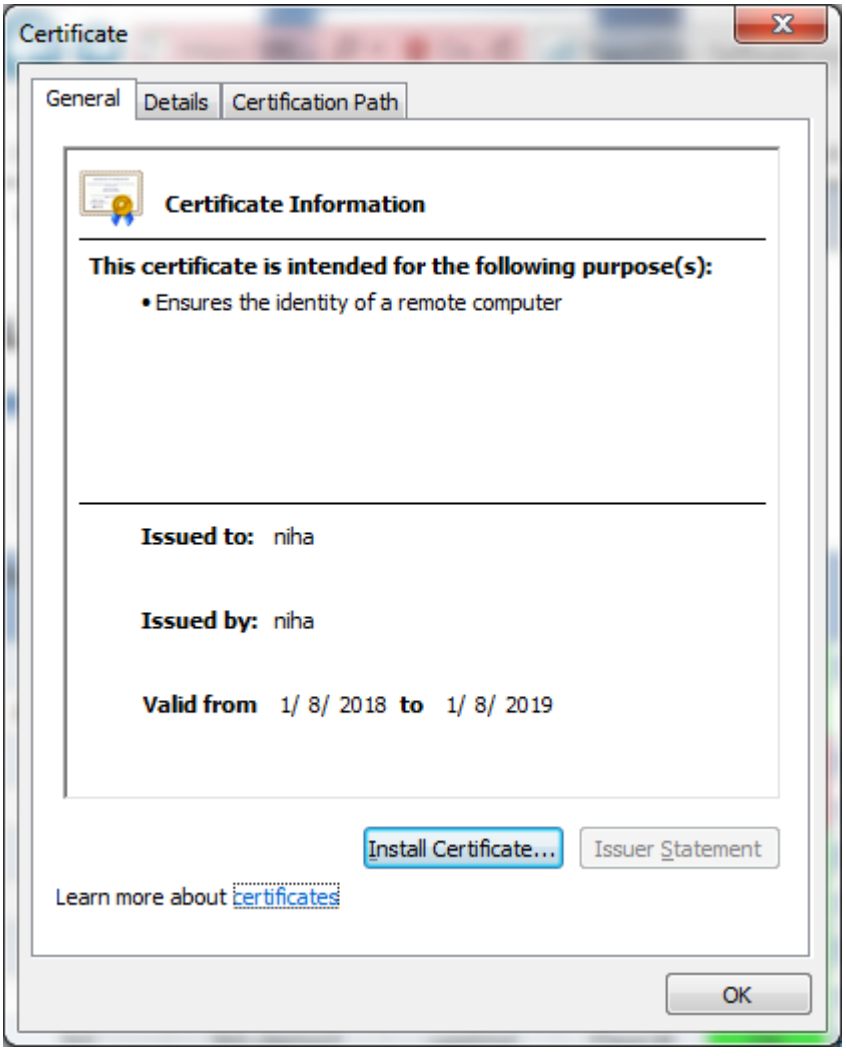
In each case, Internet Explorer will respond with a warning page about an insecure (self-signed) certificate on the web server: "There is a problem with this website's security certificate." Bypass this warning by clicking the option labeled

Continue to this website (not recommended).

anyway. The address bar will display "Certificate error" on its right side. Click on this message, and you will see a popup dialog with message "Mismatched address".

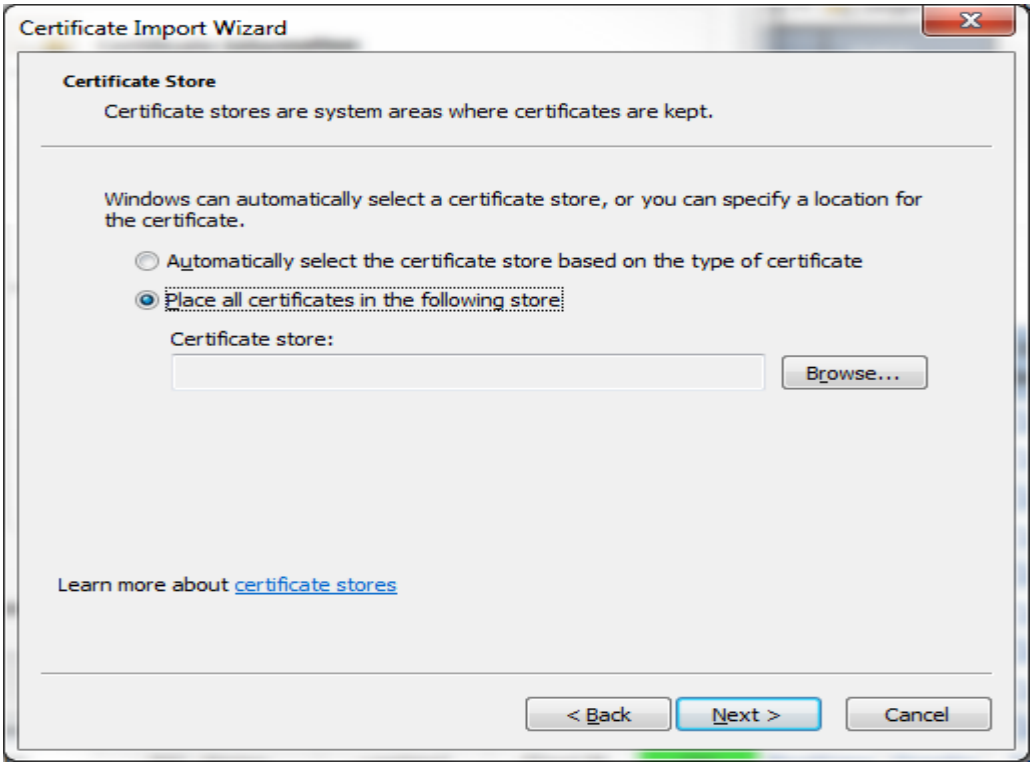


Click on the link View certificates , and a Certificate dialog will pop up.

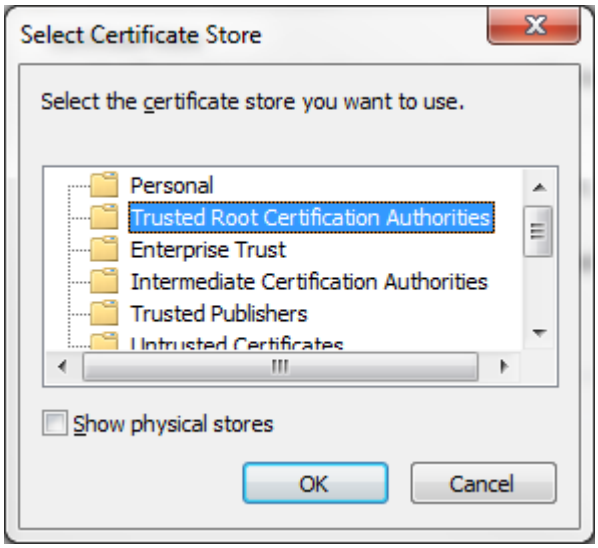


Click on the button labeled “Install Certificate...” and the Certificate Import Wizard (dialog) will pop up.

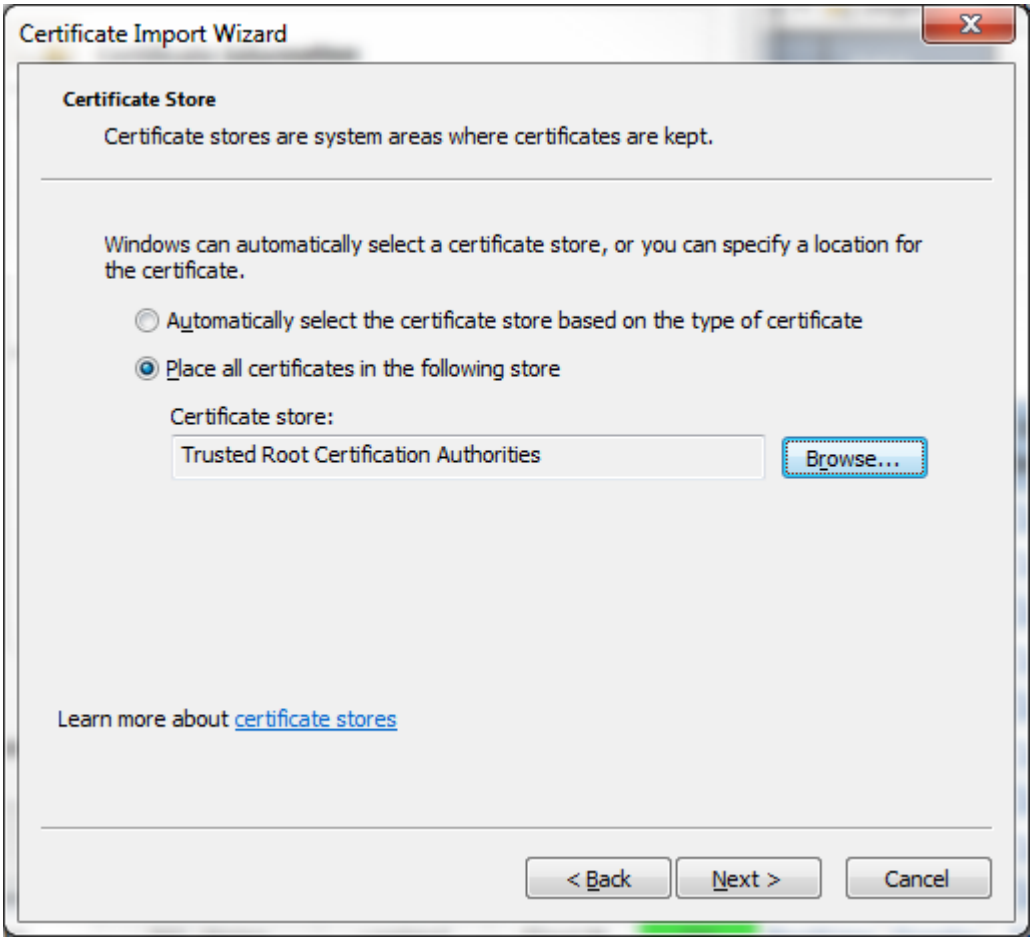
Click “Next >” button.



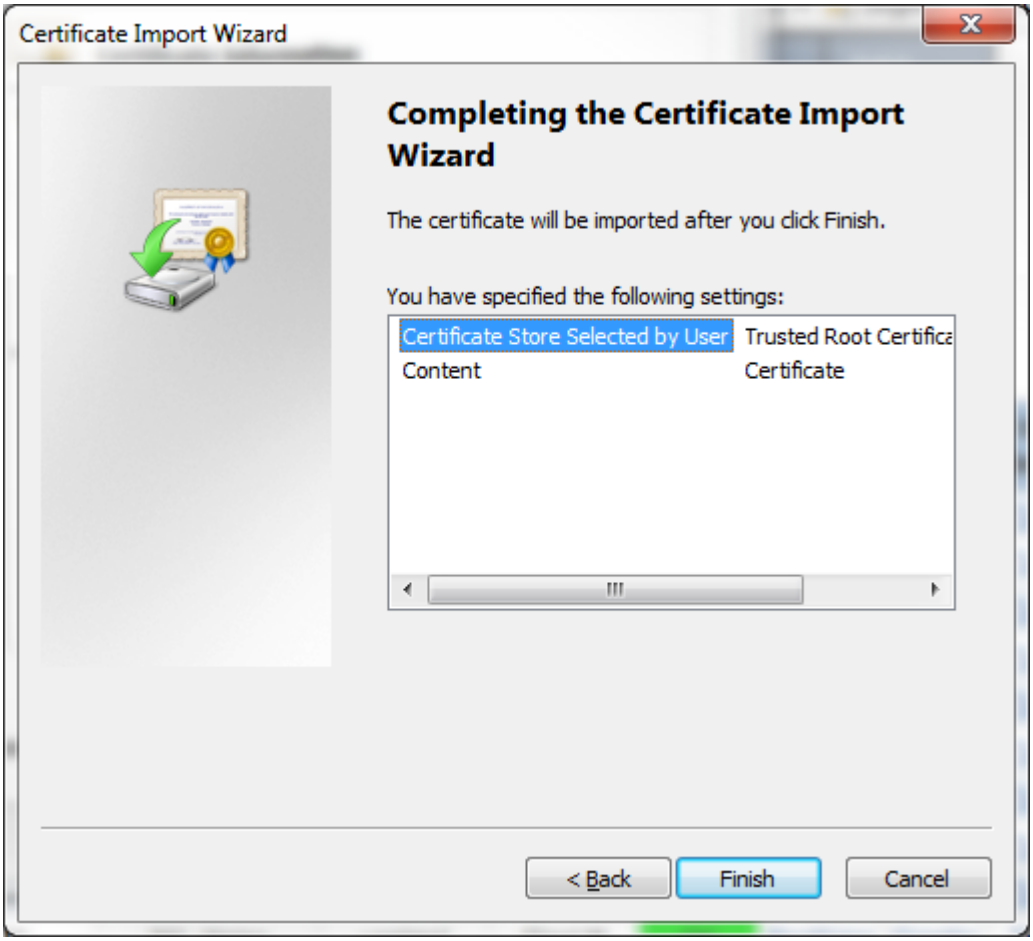
On the next page “Certificate Store” click on radio button labeled “Place all certificates in the following store” then click “Browse...” button and the Select Certificate Store dialog will pop up.



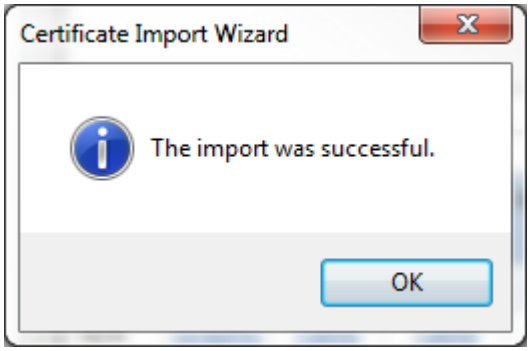
Click on the selection labeled “Trusted Root Certification Authorities” then click the OK button. You will return to the Certificate Store page with the Certificate store: field filled in with the selection.



Click “Next >” button, the next wizard page “Completing the Certificate Import Wizard” will appear.



Click the “Finish” button. The Certificate Import Wizard dialog pops up with the message “The import was successful.”

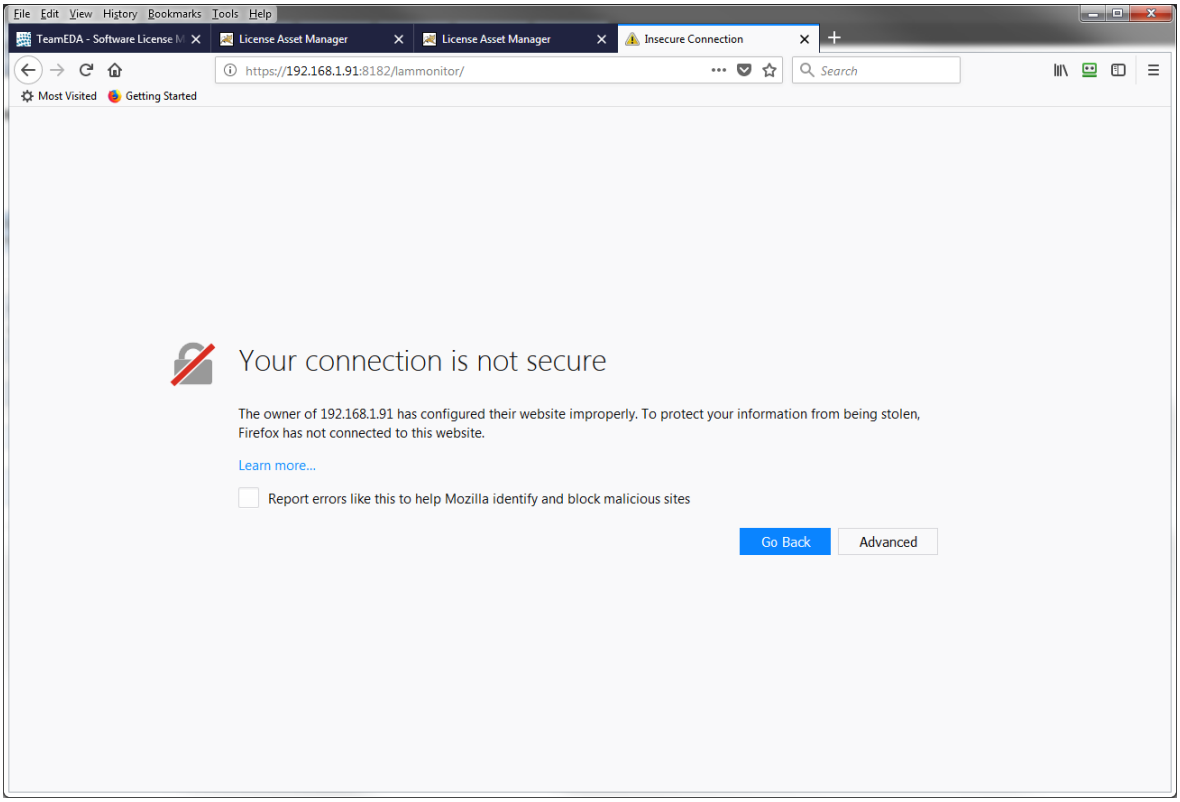


Click OK to dismiss this dialog, then OK to dismiss the Certificate dialog.

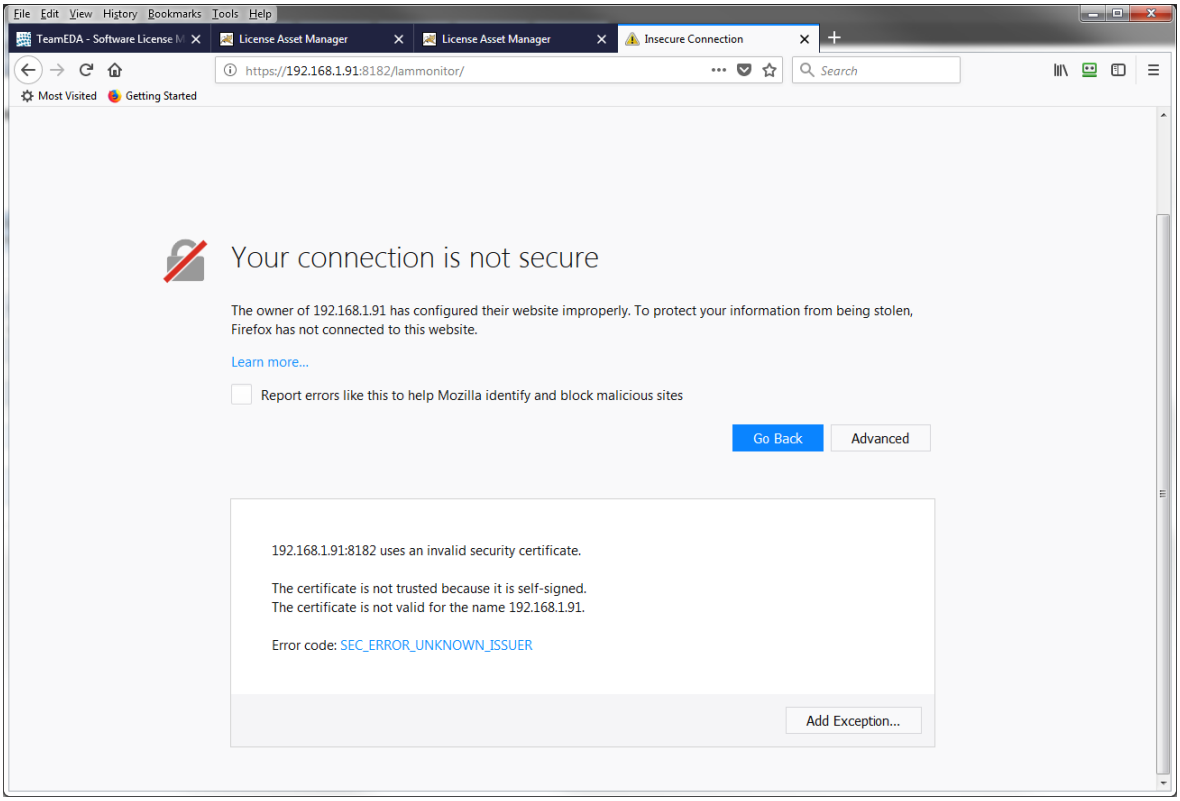
You should now be able to go back to the LAM/UM Usage or Daemon tab, refresh that page, and the proper information should now be displayed.

3c – Firefox [tested with version 57])

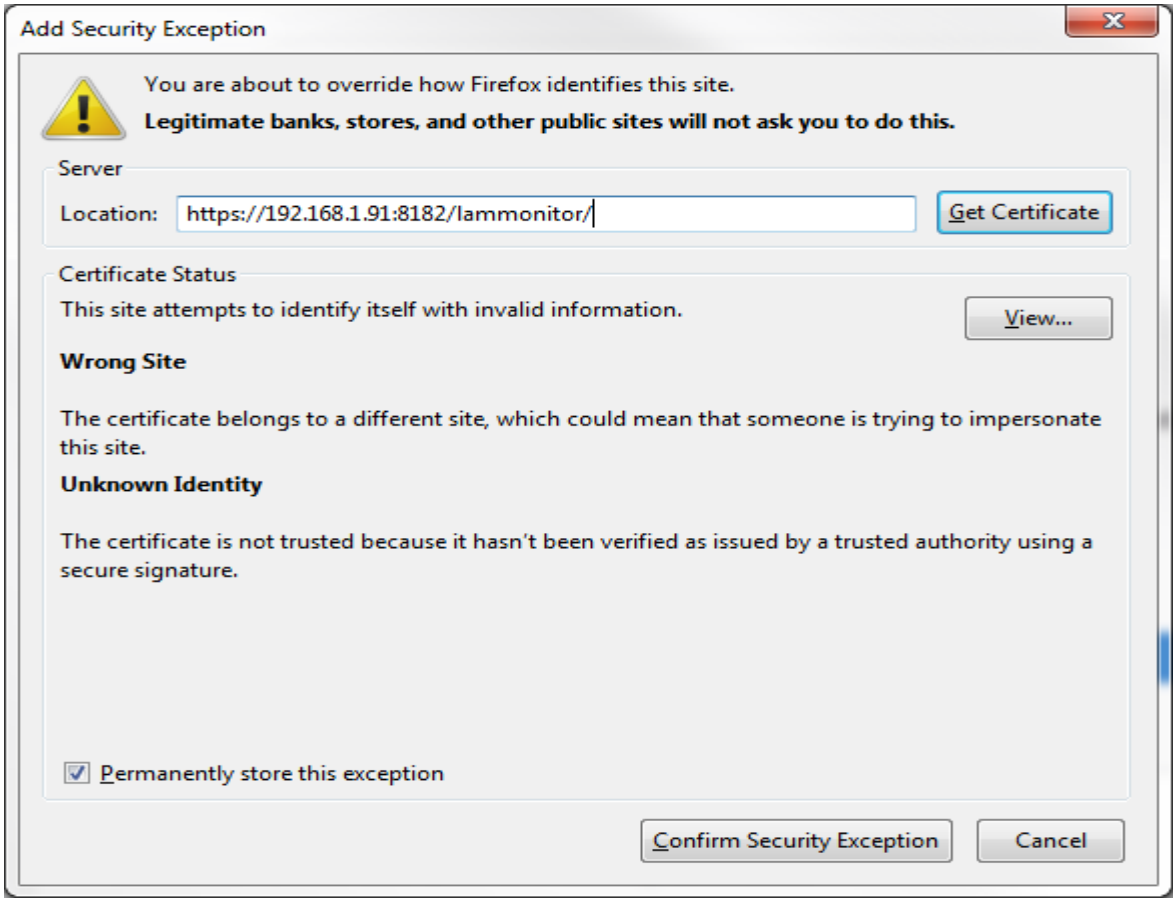
In each case, FireFox will respond with a warning page about an insecure (self-signed) certificate on the web server: “Your connection is not secure. The owner of *Your_IPaddr* has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.”



Bypass this warning by clicking the option button labeled Advanced. The page will expand with more explanation, and an option button labeled “Add Exception...”.



Click on the “Add Exception” button. You will see this dialog:



Click on "Confirm Security Exception" button. The page previously blocked will appear. You should now be able to go back to the LAM/UM Usage or Daemon tab, refresh that page, and the proper information should now be displayed.

4) At this point, all data traffic to and from your browser and LAM/UM flows over the secure (encrypted) TLS connection just configured.

If any questions:

Contact TeamEDA for help:
603-656-5200
support@teameda.com