Lexmark

Global Services

Markvision Enterprise

SSL Configuration White Paper

740 West New Circle Road
Lexington, Kentucky 40550
www.lexmark.com

Table of Contents

Document Overview

This white paper describes the steps required to secure communication between the Markvision (Tomcat) Server and a user's Web browser using the SSL (Secure Socket Layer) protocol.

Version: 3.1

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in later editions. Improvements or changes in the products or the programs described may be made at any time.

Lexmark rights of intellectual property are applicable to the document contents. The information contained herein is for the exclusive internal use for Lexmark International, Inc. and this document, or parts cannot be passed to third parties without the written agreement of Lexmark.

Trademarks

1 Overview

Secure Socket Layer (SSL) or Transport Layer Security (TLS) is a common security protocol that uses data encryption and certificate authentication to protect communication between a server and a client. For Markvision Enterprise, SSL can be used to protect sensitive information shared between the Markvision (Tomcat) Server and a user's Web browser, such as:

- Device passwords
- Security policies
- Device authentication information (LDAP, Kerberos, etc.)
- Markvision Enterprise user credentials and passwords

SSL enables the Tomcat server and the Web browser to encrypt this data before sending it and decrypt it upon receipt. SSL also requires the server to present the browser with a certificate that proves the server is who it claims to be. This certificate can either be self-signed (not recommended for production) or approved by a trusted third-party Certificate Authority (CA).

This document describes the steps required to create an SSL keystore and self-signed certificate, obtain a certificate from a CA (if desired), and configure the Tomcat server's SSL Connector. The document also explains how to configure the server to automatically redirect users to the secure Markvision Enterprise Web page (if desired).

1.1 Audience

This document is intended for server/Web administrators who are familiar with the following.

- Basic Tomcat SSL configuration
  - For detailed information, see the online documentation for Apache Tomcat 8.0 SSL Configuration HOW-TO
- Basic SSL terminology (certificate, key, keystore, alias, etc.)
- Basic keytool commands
  - Keytool is the utility used to create and manage keys and certificates. It is part of the Java Runtime Environment (JRE).

2 Configuring the Markvision Server with SSL

The Markvision Server runs as a standalone Tomcat server. You will first need to use keytool to create a keystore and an SSL certificate (either self-signed or certified by a CA). You will then need to configure the SSL Connector settings in the Tomcat server's main configuration file (server.xml).

2.1 Creating the Keystore and Self-Signed Certificate

To create a keystore containing a self-signed certificate:

1. At a command prompt, enter:
   `$MVE_INSTALL\Markvision\jre\bin\keytool -genkeypair -alias mve -keyalg RSA -keystore $MVE_INSTALLmve.jks`
   Note: *`$MVE_INSTALL` mentioned above is a placeholder. This needs to be appropriately replaced by application directory of MVE.*
2. Enter a keystore password.
3. Fill in the required information for your certificate (your name, company name, etc.).
4. When prompted, enter a key password (the password for this specific certificate). Alternatively, you may press **Enter** and set the key password to the same password as the keystore.

2.1.1 Creating the Certificate Signing Request

The steps in the previous section created a self-signed certificate that the Tomcat server will present to a user's browser to prove its identity. If you continue configuring MVE with a self-signed certificate and attempt to access MVE, you will be presented with a browser warning about the certificate:

Figure 1: Connection is Untrusted Dialog

Users will still be able to access the secure Markvision Enterprise Web page if they choose to confirm the security exception with their specific browser. To avoid the security exception, you will need to obtain a verified SSL certificate from a trusted third-party CA. To do this, you must create a Certificate Signing Request (CSR) that your CA will use to generate the verified SSL certificate. If you want to continue using a self- signed certificate, skip to Section 2.2.

1. At the command prompt, enter:
   `$MVE_INSTALL\Markvision\jre\bin\keytool -certreq -keyalg RSA -alias mve -file [my_cert_filename].csr -keystore $MVE_INSTALLmve.jks`
   Note: *`$MVE_INSTALL` mentioned above is a placeholder. This needs to be appropriately replaced by application directory of MVE.*
   Substitute your desired certificate filename for `[my_cert_filename].`
2. When prompted, enter the keystore password and, optionally, the key password if it differs from the keystore. Keytool generates a CSR file called [my_cert_filename].csr.
3. Follow the instructions on the CA's Web site to submit your CSR. The CA generates a verified SSL certificate.
4. Follow the instructions on the CA's Web site to download your certificate and the Root Certificate.

2.1.2 Importing the Certificates

Import your certificates into the keystore:

1. At a command prompt, enter:
   `$MVE_INSTALL\Markvision\jre\bin\keytool -import -alias root -keystore $MVE_INSTALLmve.jks -trustcacerts –file [root_cert_filename]`
   Substitute your Root Certificate filename for `[root_cert_filename].`
2. At a command prompt, enter:
   `$MVE_INSTALL\Markvision\jre\bin\keytool -import -alias mve -keystore $MVE_INSTALL/mve.jks -file[my_cert_filename]`
   Substitute your certificate filename for `[my_cert_filename].`

   Note : *`$MVE_INSTALL` mentioned above is a placeholder. This needs to be appropriately replaced by application directory of MVE.*

Your keystore and certificate are now certified by a trusted third-party CA.

2.2 Configuring the SSL Connector

You must configure the server to use SSL. This is done by modifying the SSL Connector settings in the server's main configuration file (server.xml).

1. Open `$MVE_INSTALLtomcatconfserver.xml` and find the SSL Connector. For example:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443
             This connector uses the JSSE configuration, when using APR, the
             connector should be using the OpenSSL style configuration
             described in the APR documentation -->
<!--
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
              sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
              SSLEnabled="true" scheme="https" secure="true" clientAuth="false"
              compression="on"
              compressableMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/
              javascript,application/json"
              maxThreads="150" maxHttpHeaderSize="16384" minSpareThreads="25"
              enableLookups="false" acceptCount="100"
              connectionTimeout="120000" disableUploadTimeout="true"
              URIEncoding="UTF-8" server="Apache"
              sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2" sslProtocol="TLS"
              keystoreFile="C:/Program Files (x86)/Lexmark/ mve.jks "
              keystorePass="keystore_password" keyAlias="mve" keyPass="key_pass"

              ciphers="TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,TLS_RSA
              _WITH_AES_128_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA"
              />
-->
```

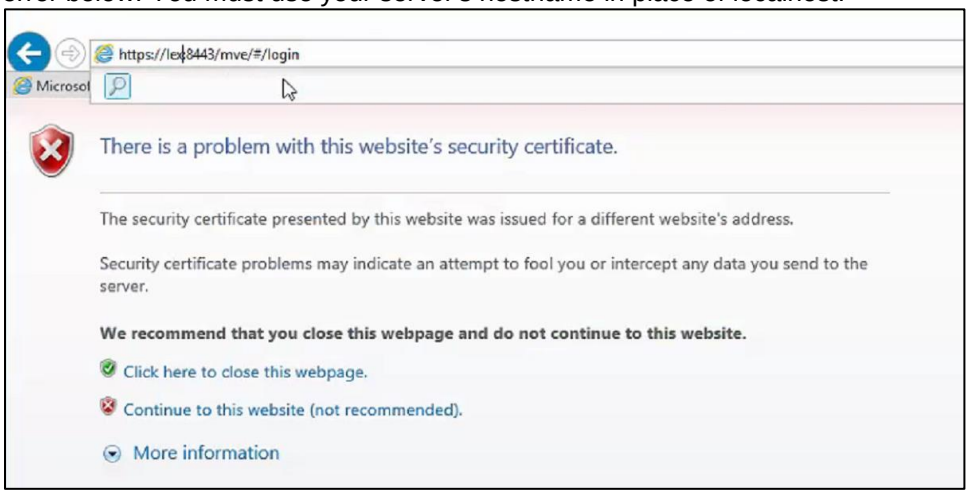
Note : *`$MVE_INSTALL` mentioned above is a placeholder. This needs to be appropriately replaced by application directory of MVE.*

2. Uncomment the SSL Connector entry. Substitute the keystore password and key password used in Section 2.1 for `keystore_password` and `key_pass`. Ensure that the Connector port= number is set to the port you plan to use for your SSL. The default is 8443.

***Note**: These cipher suites have been selected in order to address known SSL vulnerabilities such as logjam and FREAK.*

***Note**: If you placed the keystore file in a different directory, be sure to update the keystoreFile to point to the correct path as well.*

3. Restart Markvision Service.
4. Test your SSL configuration by entering the secure URL into your browser. For example, `https://localhost:8443/mve`. Hostnames matter with certificates; if you use localhost, you will get the error below. You must use your server's hostname in place of localhost.



3 Redirecting to the Secure Web Page

You can configure the server's Web.xml file to automatically redirect users to the secure Markvision Enterprise Web page when they enter the unsecure address into their browser. For example, if a user entered http://localhost:9788/mve, they would automatically be redirected to https://localhost:8443/mve.

1. Open `$MVE_INSTALL/apps/dm-mve/WEB-INF/Web.xml` and uncomment the following:

```
<!--   <security-constraint>-->
<!--         <Web-resource-collection>-->
<!--               <Web-resource-name>All Requests</Web-resource-name>-->
<!--               <url-pattern>/*</url-pattern>-->
<!--         </Web-resource-collection>-->
<!--         <user-data-constraint>-->
<!--               <transport-guarantee>CONFIDENTIAL</transport-guarantee>-->
<!--         </user-data-constraint>-->
<!--   </security-constraint>-->
```

2. Open `$MVE_INSTALL/tomcat/conf/server.xml` and ensure that the `redirectPort=` number for the non-SSL Connector matches the `Connector port=` number for the SSL Connector.

```
<Connector port="9788" maxHttpHeaderSize="16384" maxThreads="150" minSpareThreads="25"
        enableLookups="false" redirectPort="8443" acceptCount="100" connectionTimeout="120000"
        disableUploadTimeout="true" URIEncoding="UTF-8" server="Apache" compression="on"
        compressableMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/
        javascript,application/json"/>

<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"

        sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
```

3. Restart the Tomcat server.
4. Enter the unsecure address (for example, http://localhost:9788/mve) into your browser and verify that you are redirected to the secure address (for example, https://localhost:8443/mve).

Note : *`$MVE_INSTALL` mentioned above is a placeholder. This needs to be appropriately replaced with application directory of MVE.*

Lexmark International, Inc.
740 W. New Circle Road
Lexington, KY 40550, U.S.A
Tel: +1-859-232-2000
www.lexmark.com