The Keystore and a Nettlesome Certificate

Posted At : November 2, 2006 10:55 AM | Posted By : Mark Kruger
Related Categories: Coldfusion Troubleshooting

A developer on one of my lists had an issue getting a web service to work correctly. The service ran over SSL and it seemed that he was having the problem with his certificate existing in the **trusted Keystore**. He found Steve Erat's blog on using the **keytool** and followed the instructions. But the service was still giving him a handshake error(javax.net.ssl.SSLHandshakeException - no trusted certificate found). After a few more failed efforts he decided (on the advice of the list) to explore the possibility that it was a bug in the JVM based on some indicators from Sun. But he didn't want to upgrade his JVM - not wanting to have to do full-fledged regression testing. So instead, he found a nifty work around.

The Fix

Here are the steps to his fix:

1. He downloaded and installed the new Java SDK (with the new JVM version) to a separate folder.
2. He copied the file "cacerts" from the CfusionMX7/runtime/jre/lib/security to his new SDK runtime/lib/security folder. This step overwrites the "new" Keystore (from the new JVM) with the existing Keystore from the Coldfusion installation. This step is to ensure that any existing certificates being used by CF are kept in the Keystore when we copy the file back.
3. He used Internet Explorer to export the certificate in base64 to a .cer file.
4. He used the keytool against the *new* SDK with the new JVM to import it into the Keystore (the "new" Keystore).
5. He copied the cacerts file back to Cfusiionmx7/runtime/jre/lib/security/ and restarted Coldfusion.

Remember, the goal was to get a new Keystore created without touching the bug that seemed to hinder the old JVM. According to him, this seemed to work splendidly. Personally, I thought it was an innovative and time saving solution. Check it out if you have Cert issues. The talking tree article is also excellent.