**Instructions to use OpenSaml3-utils.jar**

Nationwide IT team has developed a Reference Implementation (Proof of Concept) for consumers to give an idea of generating SAML Assertion using OpenSaml 3 libraries.

**Prerequisites:**

1. Third-party/Public CA approved Certificate chain (3 .cer files)
2. A Private Key of certificate
3. Installation of OpenSSL 64 bit and add the OpenSSL installed folder to PATH
4. Installation of JDK 1.8 or higher version, and add the installed folder to JAVA\_HOME and then %JAVA\_HOME%\bin to PATH
5. OpenSAML3.2.0 librarie**s** from <https://mvnrepository.com/artifact/org.opensaml>

**Steps to Import Cert and Private Key into Java KeyStore (.JKS):**

There are many ways to add a .cer and private key into java TrustStore/KeyStore. Any approach can be used but the following steps are performed in PoC effort.

1. Create a new text file *e.g. test\_key\_cer.txt*
2. Open the private key file (e.g.: test\_privatekey.key) and copy the content and paste into test\_key\_cer.txt
3. Open the end user certificate (e.g.: test\_cert.cer) and copy the content and append into *test\_key\_cer.txt*
4. From a terminal window, launch openssl.exe. If does not work, add the OpenSSL installed folded to PATH
5. Run the following command in OpenSSL shell to export key and cert into P12 file with an alias

pkcs12 -export -in test\_key\_cer.txt -out test\_key\_cer.p12 -name ALIASNAME

1. Enter any password of your choice that will be used as one of the method arguments to a java method later. Please notice that the same password is used for simplicity.

Enter Export Password: changeit

Verifying - Enter Export Password: changeit

1. Using Java Keytool, import Keystore.  Enter the password when asked.  Please notice that the same password is used for simplicity.

keytool -importkeystore -deststorepass changeit -destkeystore C:\jdk1.8.0\_71\jre\lib\security\cacerts -srckeystore test\_key\_cer.p12 -srcstoretype PKCS12

Enter source keystore password: changeit a default password of cacerts

Output: Import command completed:  1 entries successfully imported, 0 entries failed or cancelled

1. Verify that the cert is added to truststore/keystore (cacerts) with alias name ALIASNAME

keytool -list -v -keystore C:\jdk1.8.0\_71\jre\lib\security\cacerts > C:\test.text

Enter keystore password: changeit

**Steps to use opensaml3-utils.jar in Test (non-production) environment**

1. Add *opensaml3-utils.jar*, a nationwide developed library for PoC efforts, to classpath or runtime.
2. From any Java method, invoke

Map samlMap<String, String> samlMap =  **new** OpenSaml3AssertionApplication().buildSAMLAssertionObjects("C:\\jdk1.8.0\_71\\jre\\lib\\security\\cacerts ", " changeit", " changeit", " ALIASNAME", "CN=ext-test-nwpc.nationwide.com, O=Nationwide Mutual Insurance Company, L=Columbus, ST=Ohio, C=US", "VALID\_ESS\_TEST\_USER\_OR\_SYSTEM\_ID", "<https://identity-pt.nationwide.com>", "urn:oasis:names:tc:SAML:2.0:ac:classes:Password", 60);

//Get Base64 Encoded SAML Assertion from samlMap

String base64EncodedSAMLAssertion = samlMap.get("SAML\_ASSERTION\_BASE64\_ENCODED");

**Steps to use opensaml3-utils.jar in Production environment**

1. Add *opensaml3-utils.jar*, a nationwide developed library for PoC efforts, to classpath or runtime.
2. From any Java method, invoke

Map samlMap<String, String> samlMap =  **new** OpenSaml3AssertionApplication().buildSAMLAssertionObjects("C:\\jdk1.8.0\_71\\jre\\lib\\security\\cacerts ", " changeit", " changeit", " ALIASNAME", "CN=ext-test-nwpc.nationwide.com, O=Nationwide Mutual Insurance Company, L=Columbus, ST=Ohio, C=US", "VALID\_ESS\_PROD\_USER\_OR\_SYSTEM\_ID", "<https://identity.nationwide.com>", "urn:oasis:names:tc:SAML:2.0:ac:classes:Password", 60);

//Get Base64 Encoded SAML Assertion from samlMap

String base64EncodedSAMLAssertion = samlMap.get("SAML\_ASSERTION\_BASE64\_ENCODED");