SAML Bridge Troubleshooting

For troubleshooting, it is better to run the SSO URL provided by **onelogin.saml2.idp.single_sign_on_service.url** of the **onelogin.saml.properties**, ideally, on the AD FS server.

This section provides solutions to the basic problems, such as:

Signature validation failed

If you see the following error:

ERROR c.onelogin.saml2.authn.SamlResponse - Signature validation failed. SAML Response rejected

Then it means that the public key which you referred to in **onelogin.saml.properties** is not valid, that is:

onelogin.saml2.idp.x509cert =MIIC2DCCAcCgAwIBAgIQfdRAAWmWko1IsimA004o3TANBgkqhki…

Solution

- Get a valid certificate from AD FS;
- modify onelogin.saml.properties (onelogin.saml2.idp.x509cert);
- restart Yellowfin;
- update Yellowfin SAML Bridge relying party metadata in AD FS.

top

Illegal Key Size

You may see the following exception in the Yellowfin logs:

org.apache.xml.security.encryption.XMLEncryptionException: Illegal key size

The Original Exception was **java.security.InvalidKeyException: Illegal key size**

Solution

When inspecting the SAML response payload below, the data is encrypted with AES-256:

Refer to the EncryptionMethod Algorithm here: http://www.w3.org/2001/04/xmlenc#aes256-cbc

By default, Java's key size is limited to 128-bit key due to US export laws and a few countries' import laws.

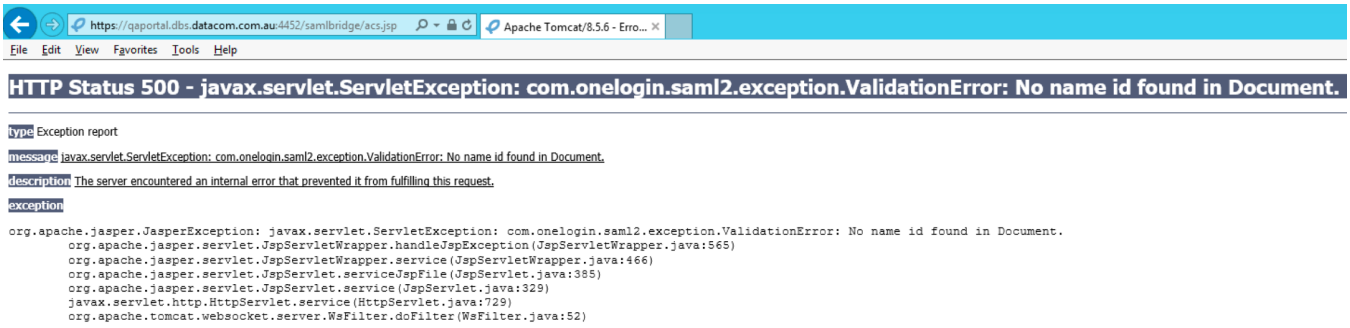
Here's a fix for this:

- Download Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files:
  Java 7: http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html
  Java 8: http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html
- Copy the **local_policy.jar** and **US_export_policy.jar** files to this directory: *[JAVA_HOME]/jre/lib/security*.

top

Invalid Name ID

SAML requires the name ID as part of the Identity Provider response. If you provide the incorrect name ID of your AD FS, then you will see the following exception in your browser:



Ensure that you pass the correct name ID and that it matches the format expected by the SAML bridge (that is, onelogin.saml2.sp.nameidformat of onelogin.saml.properties).

Here's a list of possible formats:

NAMEID_EMAIL_ADDRESS = 'urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress';

NAMEID_X509_SUBJECT_NAME = 'urn:oasis:names:tc:SAML:1.1:nameid-format:X509SubjectName';

NAMEID_WINDOWS_DOMAIN_QUALIFIED_NAME = 'urn:oasis:names:tc:SAML:1.1:nameid-format:WindowsDomainQualifiedName';

NAMEID_UNSPECIFIED = 'urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified';

NAMEID_KERBEROS = 'urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos';

NAMEID_ENTITY = 'urn:oasis:names:tc:SAML:2.0:nameid-format:entity';

NAMEID_TRANSIENT = 'urn:oasis:names:tc:SAML:2.0:nameid-format:transient';

NAMEID_PERSISTENT = 'urn:oasis:names:tc:SAML:2.0:nameid-format:persistent';

NAMEID_ENCRYPTED = 'urn:oasis:names:tc:SAML:2.0:nameid-format:encrypted';

The example below shows a list of correct Yellowfin logs regarding the SAML response:

DEBUG c.onelogin.saml2.authn.SamlResponse - SAMLResponse validated --> …

…

DEBUG c.onelogin.saml2.authn.SamlResponse - SAMLResponse has NameID --> john.smith@yellowfin.bi

DEBUG c.onelogin.saml2.authn.SamlResponse - SAMLResponse has attributes: {http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress=[john.smith@yellowfin.bi]}

DEBUG com.onelogin.saml2.SamlAuth - processResponse success --> <*very long line representing signing certificate*>

top

COULD_NOT_FIND_PERSON

If you see the following error in your Yellowfin logs:

INFO (AdministrationService:remoteAdministrationCall) - WebserviceException caught: 8(COULD_NOT_FIND_PERSON)

Then it means that you have switched off the user provision functionality, and the passed ID is not of a Yellowfin user.

top

Previous topic: Bridge operation settings
Next topic: Examples