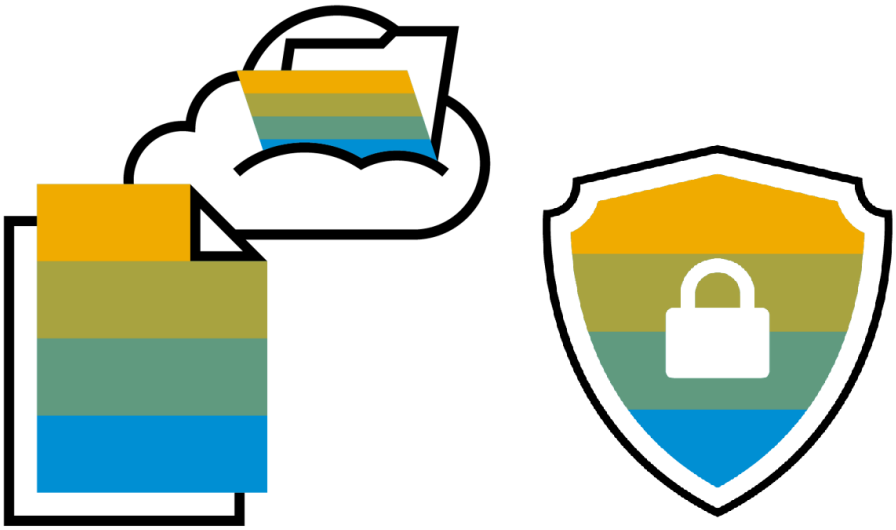
SAP BI Pattern Book Series

Pattern Book on Hybrid Analytics – Part 1
Configuring SAML-based trusted authentication between SAP BI Platform and SAP Analytics Hub



THE BEST RUN SAP

www.sap.com/contactsap

© 2018 SAP SE or an SAP affiliate company. All rights reserved.
No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See www.sap.com/copyright for additional trademark information and notices.

THE BEST RUN SAP®

Table of Contents

1 ABSTRACT 5
2 INTRODUCTION 6
2.1. What is SAP BI Pattern Books? 6
2.2. Project Overview 6
2.3. Project Scope 6
2.4. System Architecture 7
2.5. BI Landscape Technical Architecture 8
2.6. SAML Authentication Workflow 9
3. TOMCAT APPLICATION SERVER AS SAML SERVICE PROVIDER FOR BOE WEB APPLICATIONS USING SAP CLOUD PLATFORM IDENTITY PROVIDER 9
3.1 Add SAML Tomcat Service Provider jars 10
3.2 Configure Trusted Authentication with Web Session 10
3.3 Enable SSL in BI Platform 11
3.3.1 Generating keystore for Tomcat 11
3.3.2 Generating SSL certificates using GenPSE tool 11
3.3.3 Configure Tomcat to communicate with a user's browser over HTTPS 13
3.3.4 Configure Tomcat to use the SSL certificates for communication with the SIA 13
3.3.5 Configure the SIA to use the SSL certificates 14
3.4 Export Existing Users of a Tenant of SAP Cloud Platform Identity Authentication Service 15
3.5 Create Users in BOE 16
3.6 Edit the securitycontext.xml File to Enable End Points 17
3.7 Changes in config Properties for BI Platform Web Applications 17
3.8 Configurations in the Deployment Descriptor– web.xml 18
3.9 Download SAML 2.0 IdP Metadata and Update in BOE 19
3.9.1 Download the SAML 2.0 IdP metadata from SAP Cloud Platform Identity provider 19
3.9.2 Upload the SAML 2.0 IdP Metadata File to BOE 19
3.10 Generate Keystore 20
3.11 Generate and Upload the Service Provider Metadata 21
4 TOMCAT APPLICATION SERVER AS SAML SERVICE PROVIDER FOR BOE WEB APPLICATIONS USING ADFS 22
4.1 Add SAML Tomcat Service Provider jars 23
4.2 Configure Trusted Authentication with WebSession 24
4.3 Enable SSL in BI Platform 25
4.3.1 Generating keystore for Tomcat 25
4.3.2 Generating SSL certificates using GenPSE tool 25
4.3.3 Configure Tomcat to Communicate with a User's Browser Over HTTPS 26
4.3.4 Configure Tomcat to Use the SSL Certificates for Communication with the SIA 27
4.3.5 Configure the SIA to Use the SSL Certificates 27
4.4 Create Users in BOE 28
4.5 Edit the securitycontext.xml file to Enable End Points 29
4.6 Changes in config Properties for BI Platform Web Applications 29
4.7 Configurations in the Deployment Descriptor— web.xml 30
4.8 Download SAML 2.0 IdP Metadata and Update in BOE 31
4.8.1 Download the SAML 2.0 IdP metadata from ADFS Identity provider 31
4.8.2 Upload the SAML 2.0 IdP Metadata File to BOE 31
4.9 Generate Keystore 32
4.10 Export the ADFS Certificates 33

4.11 Import the ADFS Certificates Into the SP SAML Keystore 36
4.12 Generate and Upload SP Metadata 36
4.13 Import the Service Provider Metadata file in ADFS 36
5. ISSUES AND CHALLENGES 41
6. COMMON PROBLEMS 42
6.1. Presented with Logon Screens After Successful SAML Authentication 42
6.2. User Creation on BOE 43
6.3. SAML Authentication Fails When Tomcat is Behind Load Balancer or Reverse Proxy 44
6.4. How to Enable Trace Logging for BI SAML Extension (log4j) 44
7. USEFUL RESOURCES 46
7.1. SCN Blog Posts 46
7.2. SAP Notes 46
7.3. Acronyms 47

1 ABSTRACT

This pattern book covers the important hybrid workflows in configuring SAML 2.0 authentication in the SAP BI Platform for SAP Analytics Hub:

- Using SAP Cloud Platform Identity provider and BI platform as a Service Provider
- Using ADFS and BI platform as a Service Provider

Disclaimer

- This pattern book is for informational purpose only and should not be copied/reproduced without the written permission of SAP.
- The information provided in this book is based on the SAP BI Pattern Books project for a specific set of patterns/use cases executed/applied on a copy of one of our customer's actual BusinessObjects landscape within SAP labs environment. Hence, make sure to review and apply the steps that are applicable to your use cases or patterns, based on your SAP BusinessObjects BI landscape.
- Contents of this, or any related document and SAP's strategy and possible future developments, products, and or platforms directions and functionality that are discussed in this book are all subject to change and may be changed by SAP at any time for any reason without notice. Therefore, read the latest official product guides and release notes to understand the differences before executing any workflow or deploying any software components.

For further comments and questions, email to a.rajashekar@sap.com

Version History

Version	Scope	Date
1.0	First release	December 04, 2018
1.1	Second release with minor updates	February 08, 2019

2 INTRODUCTION

2.1. What is SAP BI Pattern Books?

SAP BI Pattern Books initiative is aimed at producing technical manuals with step-by-step instructions on how to deploy, configure, test, and upgrade SAP BusinessObjects BI software using a set of live examples, use cases, and patterns and how these use cases are technically executed/implemented on our customer landscapes.

2.2. Project Overview

In the previous releases, the BI Platform (more specifically the BI platform Web applications) did not support SAML-based authentication for SSO on Tomcat Application Server. Some customers, however, were implementing the custom SAML method, which involved implementing the entire Service Provider. Some other servers such as Netweaver and Weblogic have an inbuilt SAML-Service Provider module, which allows us to configure or enable SAML without having to write custom code. Since Tomcat is the default server that's been shipped with the software, it was important for us to support SAML on Tomcat.

Therefore, from BI 4.2 SP05 release onwards, BI Platform's webapps on Tomcat comes with inbuilt Service Provider (SP) implementation. This will allow customers to enable SAML-based SSO through some configuration steps and with very minimal coding (if required at all). Note that this SP implementation is done based on the third-party Spring Security SAML libraries.

2.3. Project Scope

The key objective of this pattern book project is to address the requirement of how to configure BI Platform thin clients such as BI Launchpad, OpenDocument, Fiori BI Launchpad for Single Sign-On (SSO) using SAML 2.0 for Tomcat Application Server.

Example Scenario

From SAP Analytics Cloud/SAP Analytics Hub, customers can configure Single Sign-On (SSO) to BI Platform Web applications using SAML 2.0.

This pattern book project will focus on two major workflows:

1. How can we explain and capture the important steps around integrating SAML 2.0 authentication in the BI Platform using SAP Cloud Platform Identity provider?
2. How can we explain and capture the important steps around integrating SAML 2.0 authentication in the BI Platform using ADFS

In addition to that, this book will also cover:

- The important things to know while viewing the BI Platform document link in SAP Analytics Hub
- The issues, challenges that we faced while executing this project and importantly, the useful resources that helped completing this project successfully – all in one place (yes, in one document called Pattern Book).

Note: SAML Authentication is not supported for CMC in BI 4.2 SP05 release and it is planned for a future BI release.

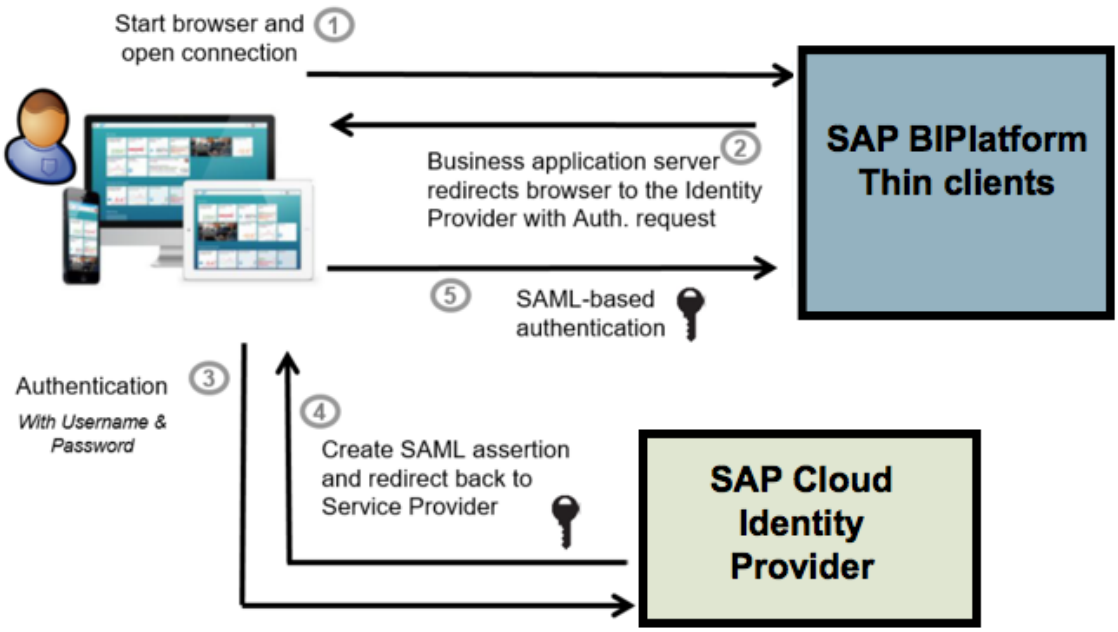
The following table summarizes the various ways the system can perform Single Sign-On (SSO) using SAML 2.0 to the applications.

Support Scope	Supported?	Support Scope
Open Document	Yes	
BI Launchpad	Yes	
Fiori BI Launchpad	Yes	
CMC	No	
Mobile Client	No	
Automatic User Sync	No	
Multiple IdPs	No	
Multiple User Types	No	
Restful Webservice Deployed on Tomcat	Yes	

2.4. System Architecture

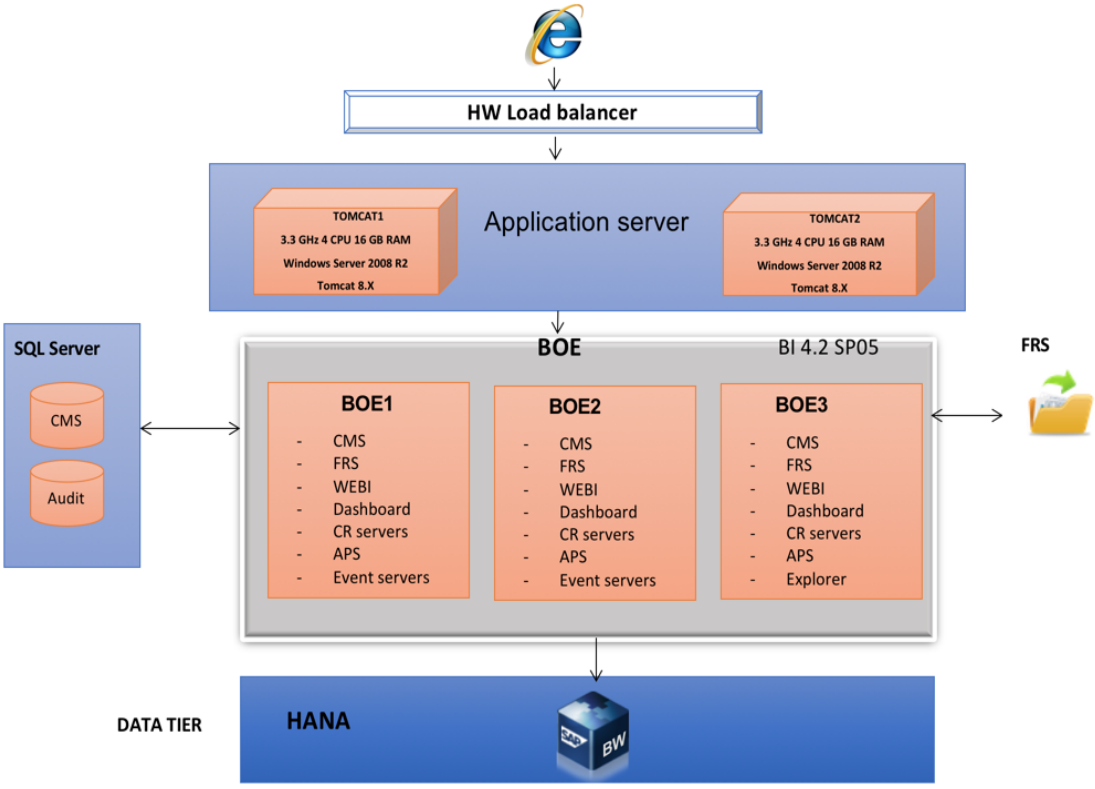
SAML 2.0 authentication in BI Platform using SAP Cloud Platform Identity provider

Before we go to the BI landscape, let us first look at how SAML 2.0 authentication in BI platform works using SAP Cloud Platform identity provider (IdP). The following diagram summarizes the sequence of steps involved in the authentication workflow.



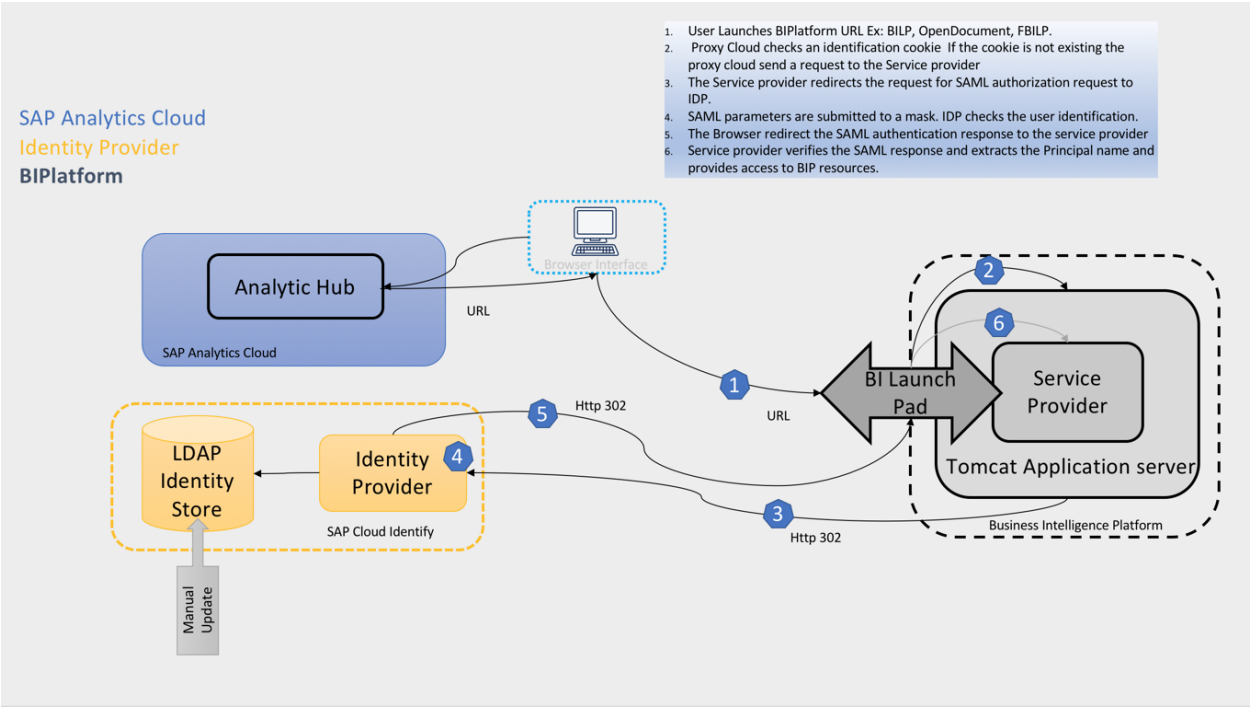
2.5. BI Landscape Technical Architecture

A two-server clustered architecture is used in this pattern book project with the separation of Web and BI platform tiers.



2.6. SAML Authentication Workflow

The following diagram explains the authentication workflow between the client tier and the backend servers and processes.



3. TOMCAT APPLICATION SERVER AS SAML SERVICE PROVIDER FOR BOE WEB APPLICATIONS USING SAP CLOUD PLATFORM IDENTITY PROVIDER

Pre-requisites

Before you configure the BI Platform Web applications for SAML 2.0 Single Sign-On using Tomcat as Application Server, you need the following:

- Install BI 4.2 SP05 on Tomcat Application Server.
- SAP Business Platform account with administrator rights.
- To integrate BI Platform with SAP Analytics Hub, we need an SAP Cloud Platform Identity provider account with administrator rights.

Note: Kindly check the supported versions of Tomcat in BI 4.2 SP05 PAM.

Summary of the Steps

Step	Action
1	Add SAML Tomcat Service Provider jars
2	Configure Trusted Authentication with Web Session
3	Enable SSL in BI Platform
4	Export Existing Users of a Tenant of SAP Cloud Platform Identity Authentication Service

Step	Action
5	Create Users in BOE
6	Edit the securitycontext.xml File to Enable End Points
7	Changes in config Properties for BI Platform Web Applications
8	Configurations in the Deployment Descriptor– web.xml
9	Download SAML 2.0 IdP Metadata and Update in BOE
10	Generate Keystore
11	Restart the Tomcat Application Server
12	Generate and Upload the Service Provider Metadata

Note: All these steps go in a sequence.

3.1 Add SAML Tomcat Service Provider jars

Note: This step is applicable only for SAML Authentication for BOE Web Applications.

Once the BOE is installed successfully, Spring SAML Service Provider JARs exist inside <BOE Install Dir>\SAP BusinessObjects Enterprise XI 4.0\SAMLJARS.

Perform the steps below to copy all these JARs into the WEB-INF\lib directory.

1. Stop Tomcat.
2. Copy these JARs to <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\lib.
3. Delete work from <BOE Install Dir>\tomcat.
4. Restart Tomcat and wait for Tomcat work to be populated.

Note: In the future releases SAML JARs will be copied automatically with the BOE default Tomcat installation. Therefore, this step will not be required.

3.2 Configure Trusted Authentication with Web Session

Though the Web server or the Web applications are being configured for SAML SSO, we rely on Trusted authentication between the Web server and the backend Central Management Server (CMS). Basically, the SP implementation would provide the user ID as extracted from the SAML assertion. This user ID is then used to log on to the CMS via trusted auth.

1. Add the global.properties file under the custom folder.

   <INSTALLDIR>\SAPBusinessObjects\tomcat\webapps\BOE\WEB-INF\config\custom.

   In case the global.properties file exists under the custom folder, the trusted authentication configuration has to be appended to the existing file.

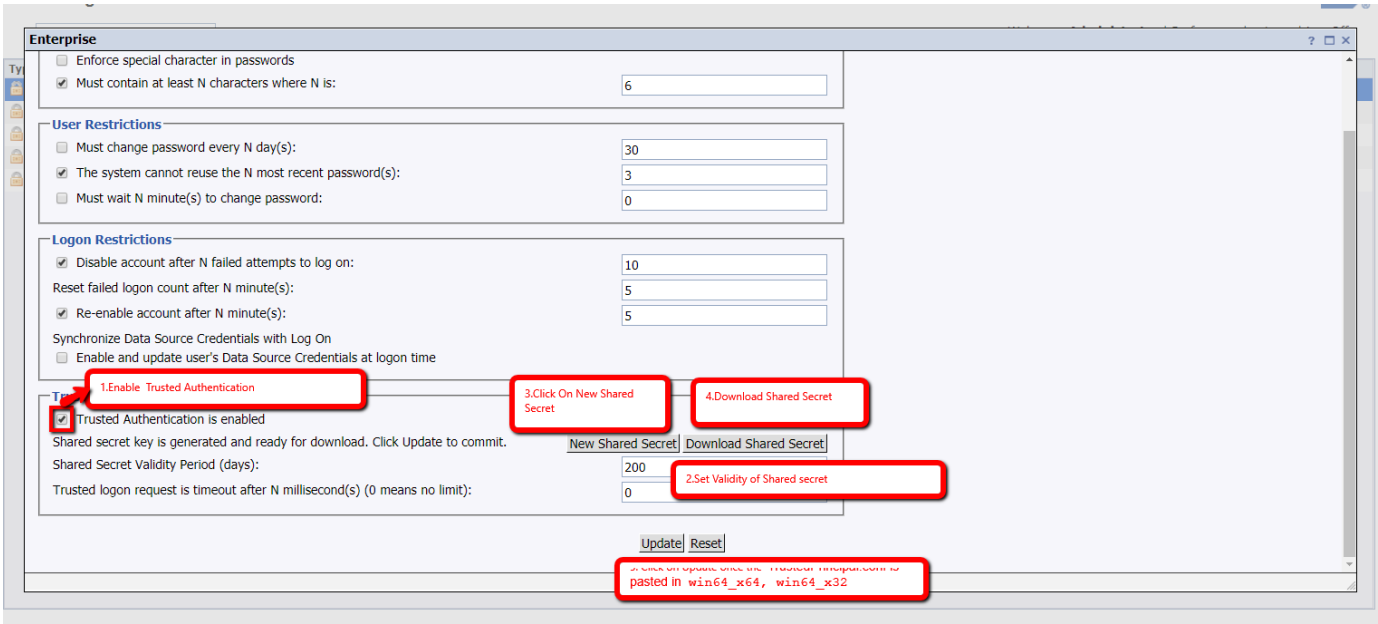
   Following is the content for global.properties:

   sso.enabled=true

   trusted.auth.user.retrieval=WEB_SESSION

   trusted.auth.user.param=UserName
2. Configure Trusted Authentication in CMC.

a. Go to CMC Application → Authentication → Enterprise. Refer to the screen below.



b. Enable Trusted Authentication.

c. Set the Validity.

d. Choose New Shared Secret.

e. To download the generated shared secret, choose Download Shared Secret.

The TrustedPrincipal.conf file is downloaded.

3. Paste the TrustedPrincipal.conf file in <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64and <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x32.
4. Go to CMC → Authentication → Enterprise and choose Update.
5. Restart Tomcat.

3.3 Enable SSL in BI Platform

Note: Steps to Enable SSL in BI Platform have changed in the BI 4.2 SP05 release.

3.3.1 Generating keystore for Tomcat

1. Navigate to: “%BOBJ INSTALL DIR%\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\jre\bin”.
2. Run commands:

   keytool.exe -genkey -alias tomcat -keysize 2048 -keyalg RSA

   MKDIR C:\SSL

   COPY "%USERPROFILE%\.keystore" C:\SSL

3.3.2 Generating SSL certificates using GenPSE tool

1. Navigate to: “%BOBJ INSTALL DIR%\SAP BusinessObjects Enterprise XI 4.0\win64_x64”.

2. Run command.

   Now, we can generate the certificate in two ways:
   - **Self-signed certificate** – CA and Server Certificates are generated using GENPSE and server certificate signing is also done using GENPSE.
   - **Generating CSR using GENPSE** – CA is generated using 3rd party library and server certificate csr using GENPSE after which, server certificate is signed by 3rd party CA using 3rd party tool (refer to section C).

3. To generate self-signed certificate, run the command: GenPSE.exe selfsigned temp.pse servercert.der cacert.der server.key passphrase.txt Default.cnf

   Note: The .cnf file should be present in the win64_x64 location, which contains default values for the certificate generation like country name, state, and so on.

4. Enter the details. By default, it will take the values from the Default.cnf file.

You must follow the following rules while creating the default configuration file:

- You should add the values on the left-hand side exactly as mentioned below.
- The values on the left-hand side are case-sensitive.
- There should be only one space between a value and the 'equal to' (=) sign. For example, there is only one space between CA_Common_Name and the 'equal to' sign.
- You must ensure there is no space after the values on the right-hand side.

5. Follow the steps below to create a default configuration file:

   a) Open a new document in a text editor.

   b) Add the values as given below:

      CA_Common_Name = rootnm

      CA_Country = DE

      CA_State = BW

      CA_Locality = RRR

      CA_Email = root@gmail.com

      CA_Unit = root_u

      CA_Expiration[YYMMDD] = yymmdd

      User_Expiration[YYMMDD] = yymmdd

      User_Country = IN

      User_State = KA

      User_Locality = BLR

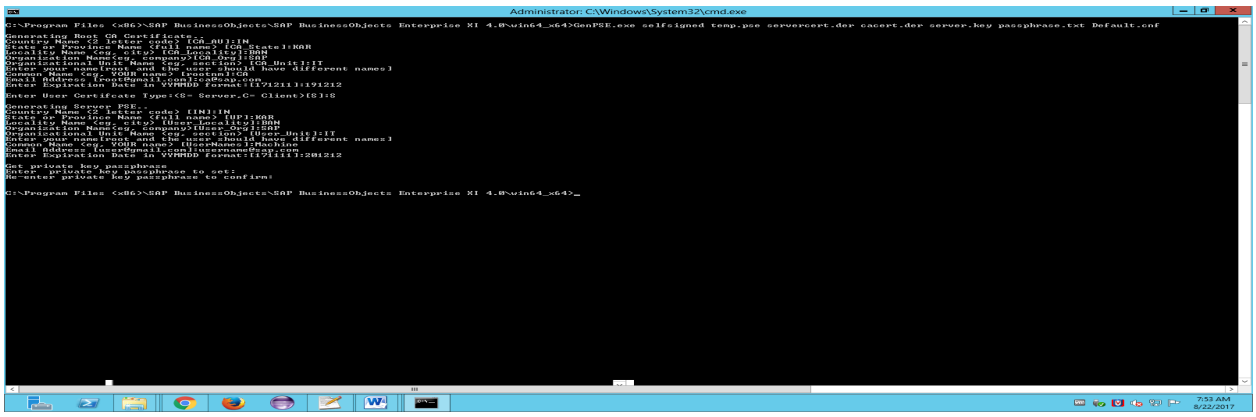
      User_Organization = SSS

      User_Unit = Unit

      User_Common_Name = UserName

   c) Save the file at <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64 with default.cnf name.

   d) Make sure that Root CA certificate and PSE files are given different Common names.



After the above command is run, the following five files are created.

cacert.der

servercert.der

server.key

passphrase.txt

temp.pse

6. Place the above files in C:\SSL

   COPY cacert.der C:\SSL

   COPY servercert.der C:\SSL

   COPY server.key C:\SSL

   COPY temp.pse C:\SSL

   COPY passphrase.txt C:\SSL

3.3.3 Configure Tomcat to communicate with a user's browser over HTTPS

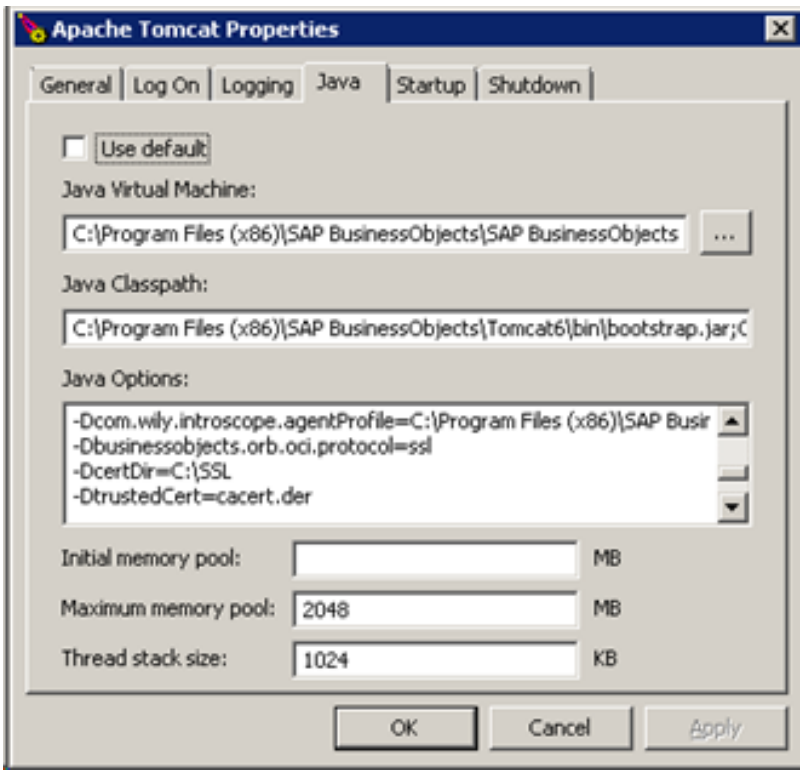
1. Open Central Configuration Manager (CCM).
2. Stop Tomcat.
3. Navigate to server.xml path (%BOBJ INSTALL DIR%\tomcat\conf), keep a copy of server.xml.
4. Edit the server.xml file and search tag with port 8080.
5. Add the below statement after the 8080 port tag.

   <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true" maxThreads="150" scheme="https" secure="true" clientAuth="false" sslProtocol="TLS" minSpareThreads="25" maxSpareThreads="75" enableLookups="false" disableUploadTimeout="true" acceptCount="100" debug="0" keystorePass="Password1" keystoreFile="C:\SSL\.keystore"/>

6. Save and close the server.xml file.

3.3.4 Configure Tomcat to use the SSL certificates for communication with the SIA

1. Open Tomcat configuration.
2. Go to the Java tab.



3. Add the text given below in the Java Options field.

   -Dbusinessobjects.orb.oci.protocol=ssl

   -DcertDir=C:\SSL

   -DtrustedCert=cacert.der

   -DsslCert=servercert.der

   -DsslKey=server.key

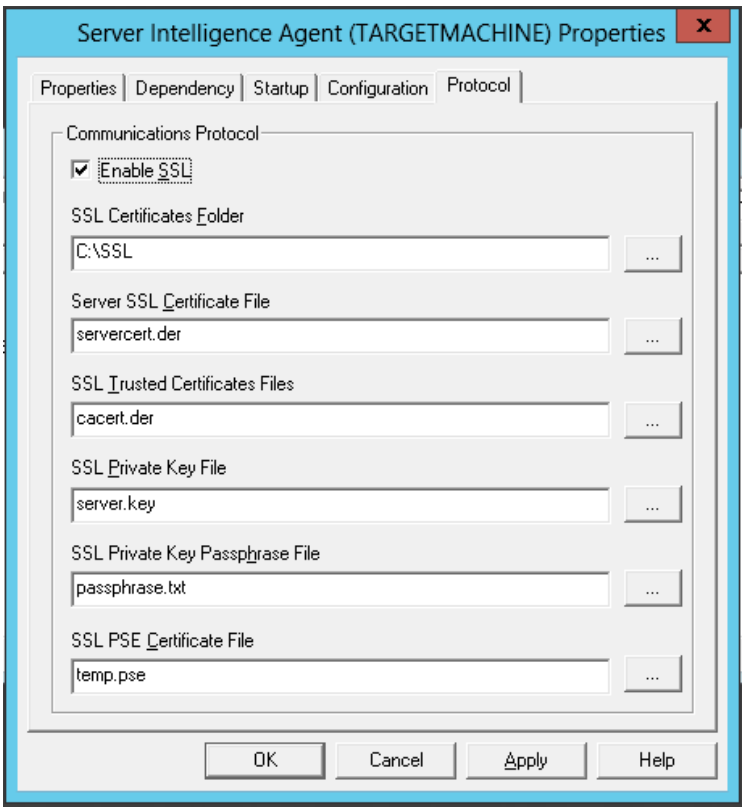
   -Dpassphrase=passphrase.txt

   Note: Do not include a space at end or beginning, otherwise, Tomcat won't start.

4. Click OK to start Tomcat again.

3.3.5 Configure the SIA to use the SSL certificates

1. In the CCM, stop the Server Intelligence Agent.
2. Double-click SIA and go to the Protocol tab.
3. Select Enable SSL.
4. Browse all files.



5. Click OK to start SIA.

   It should now be accessible using https://Servername(localhost):8443/BOE/CMC.

6. For setting SSL parameters, run the command:

   sslconfig.exe -dir C:/SSL -mycert servercert.der -rootcert cacert.der -mykey server.key -passphrase passphrase.txt -psecert temp.pse -protocol ssl

3.4 Export Existing Users of a Tenant of SAP Cloud Platform Identity Authentication Service

You can download a CSV file containing information of up to 10,000 tenant users in SAP Cloud Platform Identity authentication service including the tenant administrators. The CSV file contains the following columns: status, loginName, mail, firstName, and lastName. If the status of a user is inactive, he or she cannot perform any operations on the tenant.

Example

A tenant administrator downloads a CSV file with the current users in the system. As a result, the administrator receives the following information.

Status	LoginName	E-mail	firstName	LastName
active	EID00001	User1.Name@example.com	User1	Name
active	EID00002	User2.Name@example.com	User1	Name
active	EID00003	User3.Name@example.com	User1	Name
active	EID00004	User4.Name@example.com	User1	Name
active	EID00005	User5.Name@example.com	User1	Name

Status	LoginName	E-mail	firstName	LastName
Inactive	EID00006	User6.Name@example.com	User1	Name

For more details, refer to the following link:

https://help.sap.com/viewer/6d6d63354d1242d185ab4830fc04feb1/Cloud/en-US/40c29d2632b744af9bc7b7d353616d52.html

All users but one can log on to tenant applications. User6 cannot log on because his user is not active.

To export tenant users from Identity authentication, proceed as follows:

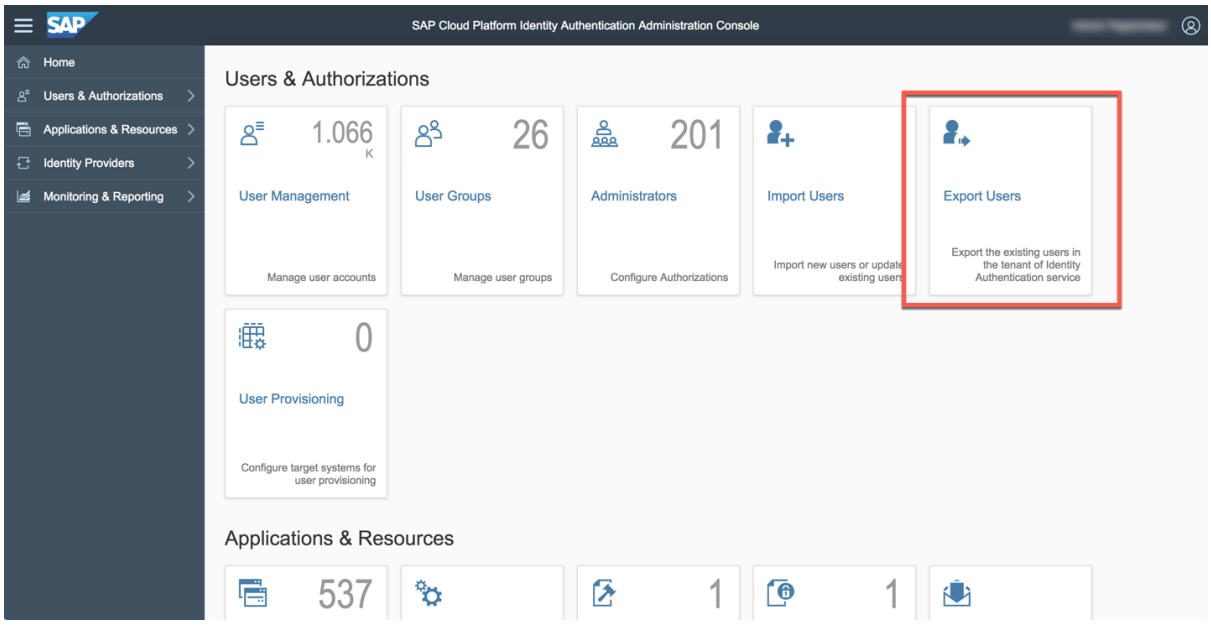
1. Access the tenant's administration console for SAP Cloud Platform Identity Authentication service by using the console's URL.

   Note: The URL has https://<tenantID>.accounts.ondemand.com/admin pattern. Tenant ID is an automatically generated ID by the system. The first administrator created for the tenant receives an activation e-mail with a URL in it. This URL contains the tenant ID.

2. Choose the Export Users tile.

   This operation opens the Export Users page.

3. Choose the Export button.



3.5 Create Users in BOE

Now that we have the list with details of the users from the IdP, we need to create/import them as Enterprise users in BOE from this SAP Cloud Platform Identity provider.

Users can be imported into BOE through the CSV file or by using an SDK script.

Note: The SAML-based authentication relies on Trusted Authentication from the Web server to the CMS. For this, the IdP users will have to be created in BOE as Enterprise users.

Please refer to the following link for importing bulk users from CMC:

https://blogs.sap.com/2013/05/16/bi-40-sp6-how-to-import-users-in-bulk-from-central-management-console/

3.6 Edit the securitycontext.xml File to Enable End Points

The securityContext.xml file is located at <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF. In the securityContext.xml file, locate the SAML entry point in the XML code as below.

Please see the section below:

```
<security:http entry-point-ref="samlEntryPoint" use-expressions="false">
<!– Comment/Uncomment for Launchpad–>
<security:intercept-url pattern="/BI" access="IS_AUTHENTICATED_FULLY"/>
<!– Uncomment for Opendocument–>
<!–<security:intercept-url pattern="/OpenDocument/**" access="IS_AUTHENTICATED_FULLY"/>–>
<!– Uncomment for Fiori Launchpad–>
<!–<security:intercept-url pattern="/BILaunchpad" access="IS_AUTHENTICATED_FULLY"/>–>
<security:custom-filter before="FIRST" ref="metadataGeneratorFilter"/>
<security:custom-filter after="BASIC_AUTH_FILTER" ref="samlFilter"/>
</security:http>
```

Examples to configure for BI Platform Web applications

- For BI Launchpad, by keeping this line uncommented <security:intercept-url pattern="/BI" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

- For OpenDocument, by keeping this line uncommented <security:intercept-url pattern="/OpenDocument/**" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

- For Fiorified BI Launchpad, by keeping this line uncommented <security:intercept-url pattern="/BILaunchpad" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

Note: The XML tag for Classical BI Launchpad is enabled by default.

3.7 Changes in config Properties for BI Platform Web Applications

A new property "saml.enabled =true" has to be added in the respective BI Platform Web application config files.

As in any other properties' setting, it is recommended to put this property in the /config/custom/<application>.properties file.

If you do not already have any custom property file here, please create an empty <application>.property. To be sure, refer to the exact name in the /config/default directory

For example

(Assuming custom properties file does not exist. If it already does, append the property **saml.enabled=true**)

- For Classic BI Launchpad, create BIlaunchpad.properties under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

- For Fiorified BI LaunchPad, create fioriBI.properties under under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

- For OpenDocument, create OpenDocument.properties under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

Add **saml.enabled =true**.

Note: It is mandatory to uncomment the specific endpoint and also add saml.enabled=true properties in custom properties file for the respective webapp, to enable SAML Authentication.

3.8 Configurations in the Deployment Descriptor– web.xml

A new filter has been introduced for SAML 2.0. The relevant section in the web.xml will be kept commented by default. Enable filters in web.xml of BOE webapp by uncommenting the SAML section(s).

Web.xml file path: <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF web.xml.

Uncomment the sections that have SAML comment as shown in the following images.

1. Uncomment the listener and context param.

   Commented listener and context param

```
        <!--SAML-->
    <!--<context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>
            /WEB-INF/securityContext.xml
        </param-value>
    </context-param>-->
    <!--SAML-->
    <!--SAML-->
    <!--<listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>-->

    <!--SAML-->
```

   After uncommenting the listener and context param the web.xml file looks as follows.

```
    <!--SAML-->
<context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>
        /WEB-INF/securityContext.xml
    </param-value>
</context-param>
<!--SAML-->
<!--SAML-->
<listener>
<listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>
```

2. Uncomment the SAML filters and mapping.

   Commented SAML filters and mapping

```
    <!--SAMl-->
<!--<filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
    </filter>

<filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
        <dispatcher>REQUEST</dispatcher>
        <dispatcher>FORWARD</dispatcher>
    </filter-mapping>-->
    <!--SAML ends -->
```

After uncommenting the SAML filters and mapping

```
    <!--SAMl-->
<filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
    </filter>

<filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
        <dispatcher>REQUEST</dispatcher>
        <dispatcher>FORWARD</dispatcher>
    </filter-mapping>
    <!--SAML ends -->
```

3. Save the web.xml with these changes.

3.9 Download SAML 2.0 IdP Metadata and Update in BOE

3.9.1 Download the SAML 2.0 IdP metadata from SAP Cloud Platform Identity provider

1. Access the tenant's administration console for SAP Cloud Platform Identity Authentication service by using the console's URL.

   Note: The URL is of the form: https://<tenantID>.accounts.ondemand.com/admin. Tenant ID is an automatically generated ID by the system. The first administrator created for the tenant receives an activation e-mail with a URL in it. This URL contains the tenant ID.

2. Choose the Tenant Settings tile.
3. Choose the SAML 2.0 Configuration list item.

   The SAML 2.0 Configuration page that opens displays the name of the Identity provider, its endpoints, and its signing certificate.

4. You can choose between the download options.
5. To download the Identity provider's metadata, press the Download Metadata File button.

For more details, please refer to the following link:
https://help.sap.com/viewer/6d6d63354d1242d185ab4830fc04feb1/Cloud/en-US/e81a19b0067f4646982d7200a8dab3ca.html

3.9.2 Upload the SAML 2.0 IdP Metadata File to BOE

1. Rename the file to idp-meta-downloaded.xml.
2. Copy the downloaded metadata file to <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF and rename it to idp-meta-downloaded.xml.

Note: By default, "idp-meta-downloaded.xml" name is generated in the securityContext.xml. We have to maintain the same name for the IdP metadata xml.

If BOE is deployed on a non-windows Platform, the path separators in the file path to the IdP metadata under the bean FilesystemMetadataProvider should be changed in securityContext.xml (<BOE Install Dir>\tomcat\webapps\BOE\WEB-INF).

That is, <value type="java.io.File">/WEB-INF/idp-meta-downloaded.xml</value> has to be changed to <value type="java.io.File">\WEB-INF\idp-meta-downloaded.xml</value> for Linux.

```
                <constructor-arg>
<bean class="org.opensaml.saml2.metadata.provider.FilesystemMetadataProvider">
                <!-- URL containing the metadata -->
                <constructor-arg>
                    <value type="java.io.File">file:///C:/idp-meta-downloaded.xml</value>-->
                    <value type="java.io.File">\WEB-INF\idp-meta-downloaded.xml</value>
                </constructor-arg>

                <property name="parserPool" ref="parser
            </bean>
                </constructor-arg>
                <constructor-arg>
                    <bean class="org.springframework.security.saml.metadata.ExtendedMetadata">
                    </bean>
                </constructor-arg>
                <property name="metadataTrustCheck"  value="false"/>
            </bean>
            <!-- Example of HTTP metadata without Extended Metadata -->
```

Path seperators changed to /WEB-INF/idp-meta-downloaded.xml (for linux) from \WEB-INF\idp-meta-downloaded.xml (forwindows)

3.10 Generate Keystore

Note: This step is optional and is applicable only if you want to use your own keystore file.

SAML exchanges involve usage of cryptography for the signing and encryption of data. A sample self-signed keystore sampletestKeystore.jks is packaged with the product and is valid until October 18, 2019. The sampletestKeystore.jks file has an alias name Testkey and password Password1. You can now generate a self-signed keystore file using the JAVA utility keytool.

Follow the steps below to generate a keystore file:

1. Navigate to <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\bin
2. Run the command: keytool -genkeypair -alias aliasname -keypass password -keystore samplekeystore.jks -validity numberofdays

Example

keytool -genkeypair -alias TestAlias -keypass AliasPassword -keystore sampleKeystore.jks -validity 735

Command	Description
-alias	Enter the alias name of the certificate
-keypass	Enter the certificate's password
-keystore	Name of the keystore file
-validity	Validity of the certificate
numberofdays	Number of days for which the self-signed certificate is valid

The following questions are prompted after executing the command:

- Enter keystore password: *****(Password1)
- Re-enter new password: *****(Password1)
- What is your first and last name?: <Name>
- What is the name of your organizational unit?: BusinessObjects
- What is the name of your organization?: SAP
- What is the name of your city and locality?: <CityName>
- What is the name of your State and Province?: <ProvinceName>
- What is the two-letter country code for this unit?: <CountryCode>

3. Stop the Tomcat Application Server.

   The keystore file is generated at <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\bin.

4. Move the keystore file to <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF.
5. Edit the xml file located at <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF with the new alias name, password, and keystore file name.
6. Refer to the XML code below:

   <bean id="keyManager" class="org.springframework.security.saml.key.JKSKeyManager"><constructor-arg value="/WEB-INF/sampleKeystore.jks"/><constructor-arg type="java.lang.String" value="Password1″/><constructor-arg><map><entry key=" TestAlias " value="AliasPassword"/></map></constructor-arg><constructor-arg type="java.lang.String" value=" TestAlias "/></bean>

Refer to the table below for understanding the arguments.

Tag	Description
<constructor-arg value="/WEB-INF/sampleKeystore.jks"/>	Locates the keystore file
<constructor-arg type="java.lang.String" value="Password1″/>	Password for the keystore file
<entry key=" TestAlias " value=" AliasPassword"/>	Alias password
<constructor-arg type="java.lang.String" value=" TestAlias "/>	Alias of the default certificate
numberofdays	Number of days for which the self-signed certificate is valid

Note: SP metadata has to be generated every time this keystore file is changed. Our sample sp metadata will work only with our sample keystore certificate.

For more details, please refer to the following link:

https://help.sap.com/viewer/2e167338c1b24da9b2a94e68efd79c42/4.2.5/en-US/1c142c24a4384014bd10e1ea6b724c81.html

7. Restart the Tomcat Application Server.

3.11 Generate and Upload the Service Provider Metadata

1. Go to any browser and launch http://host:port/BOE/BI/saml/metadata.
   The XML file gets downloaded automatically after navigating to the above URL.

2. Upload the XML file to the SAP Cloud Platform Identity provider.

   Note: You can use the default Service Provider metadata file spring_saml_metadata.xml located at<INSTALLDIR>\tomcat\webapps\BOE\WEB-INF instead of generating it manually.

   You must replace the XML tag <replace_withip> with the IP address of the machine and <replace_withport> with the port number of the Tomcat Application Server. Replace HTTP with HTTPS if you have enabled HTTPS in Tomcat.

3. Access the tenant's administration console for SAP Cloud Platform Identity authentication service by using the console's URL.

   Note: The URL is of the following format: https://<tenantID>.accounts.ondemand.com/admin Tenant ID is an automatically generated ID by the system. The first administrator created for the tenant receives an activation e-mail with a URL in it. This URL contains the tenant ID.

4. Choose the Applications tile.

   This operation opens a list of the applications.

5. Choose the +Add button on the left-hand panel to add a new application to the list.
6. Choose the Trust tab.
7. Under SAML 2.0, choose SAML 2.0 Configuration.
8. Upload the Service Provider metadata XML file or manually enter the communication settings negotiated between Identity authentication and the Service Provider.
9. Restart the Tomcat Application Server.

Note: To check if SAML integration is successful, once you launch the SAML configured application (BI launchpad, Fiorified BI launchpad or OpenDocument), you are redirected to the IdP.

For more details please refer to the following links:

https://help.sap.com/viewer/6d6d63354d1242d185ab4830fc04feb1/Cloud/en-US/f96e4c5930a94d1ba117e05a3f3c30fc.html

https://help.sap.com/viewer/6d6d63354d1242d185ab4830fc04feb1/Cloud/en-US/be6d6f210d30404d827f8c9e78ec4489.html

4 TOMCAT APPLICATION SERVER AS SAML SERVICE PROVIDER FOR BOE WEB APPLICATIONS USING ADFS

Pre-requisites

Before you configure the BI Platform Web applications for SAML 2.0 Single Sign-On using Tomcat as the Application Server, you need the following:

- BI 4.2 SP05 installed on Tomcat Application Server.
- SAP Business Intelligence Platform account with administrator rights.
- ADFS IdP account with administrator rights.
- ADFS successfully installed and configured.

1. To verify the ADFS functionality, log on to the Windows machine using AD user and open the IE browser and type: https://adfs.adatum.com/federationmetadata/2007-06/federationmetadata.xml and then verify that the file loads successfully.
2. Configure Trusted authentication with WebSession.
3. Configure https for Tomcat and for all BI Platform Web applications.
4. Create/Import the Enterprise users in BI Platform.

Note: If you are getting a page cannot be displayed error:

- Please try and disable the proxy settings in your VM.
- Check the supported versions of Tomcat in BI 4.2 SP05 PAM.
- Verify if the Enterprise user names are the same as in the IdP.

Summary of the Steps

Step	Action
1	Add SAML Tomcat Service Provider jars
2	Configure Trusted Authentication with WebSession
3	Enable SSL in BI Platform
4	Create Users in BOE
5	Edit the securitycontext.xml file to Enable End Points
6	Changes in config Properties for BI Platform Web Applications
7	Configurations in the Deployment Descriptor— web.xml
8	Download SAML 2.0 IdP Metadata and Update in BOE
9	Generate Keystore
10	**Error! Reference source not found.**
11	Export the ADFS Certificates
12	Import the ADFS Certificates Into the SP SAML Keystore
13	Generate and Upload SP Metadata
14	Import the Service Provider Metadata file in ADFS

Note: All these steps go in a sequence.

4.1 Add SAML Tomcat Service Provider jars

Note: This step is applicable only for SAML Authentication for BOE Web Applications.

Once the BOE is installed successfully, Spring SAML Service Provider JARs exist inside <BOE Install Dir>\SAP BusinessObjects Enterprise XI 4.0\SAMLJARS.

Perform the steps given below to copy all these JARs into the WEB-INF\lib directory.

1. Stop Tomcat.
2. Copy these JARs to <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\lib.
3. Delete work from <BOE Install Dir>\tomcat.
4. Restart Tomcat and wait for Tomcat work to be populated.

Note: In the future releases SAML JARs will be copied automatically with the BOE default Tomcat installation. Therefore, this step will not be required.

4.2 Configure Trusted Authentication with WebSession

Though the Web server or the Web applications are being configured for SAML SSO, we rely on trusted authentication between the Web server and the backend Central Management Server (CMS). Basically, the SP implementation would provide the user ID as extracted from the SAML assertion. This user ID is then used to log on to the CMS via trusted auth.

1. Add the global.properties file under the custom folder.

   <INSTALLDIR>\SAPBusinessObjects\tomcat\webapps\BOE\WEB-INF\config\custom.

   In case global.properties file exists under the custom folder, the trusted authentication configuration has to be appended to the existing file.

   Following is the content for global.properties:

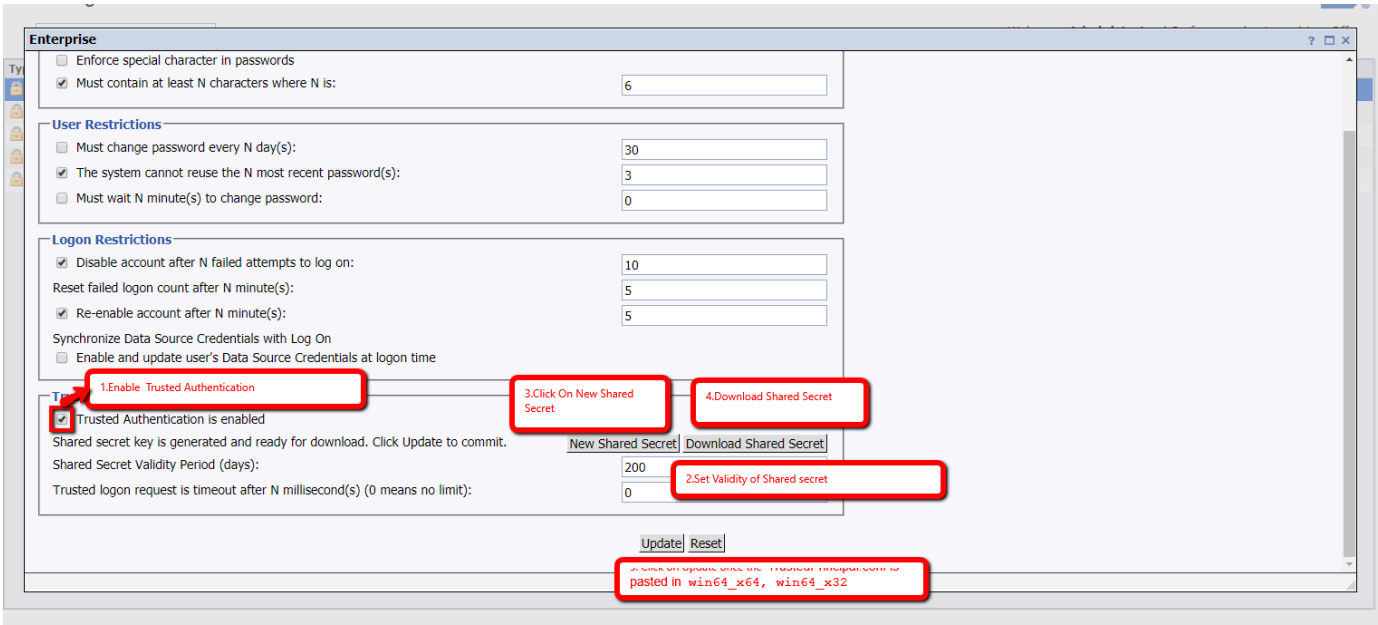
   sso.enabled=true

   trusted.auth.user.retrieval=WEB_SESSION

   trusted.auth.user.param=UserName

2. Configure Trusted Authentication in CMC.

   a. Go to CMC Application → Authentication → Enterprise. Refer to the screen below.



   b. Enable Trusted Authentication.

   c. Set the Validity.

   d. Choose New Shared Secret.

   e. To download the generated shared secret, choose Download Shared Secret.

      The TrustedPrincipal.conf file is downloaded.

3. Paste the TrustedPrincipal.conf file in <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64and <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x32.

4. Go to CMC → Authentication → Enterprise and choose Update.

5. Restart Tomcat.

4.3 Enable SSL in BI Platform

Note: Steps to Enable SSL in BI Platform is changed in the BI 4.2 SP05 release.

4.3.1 Generating keystore for Tomcat

1. Navigate to: "%BOBJ INSTALL DIR%\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\jre\bin"
2. Run commands:

   keytool.exe -genkey -alias tomcat -keysize 2048 -keyalg RSA

   MKDIR C:\SSL

   COPY "%USERPROFILE%\.keystore" C:\SSL

4.3.2 Generating SSL certificates using GenPSE tool

1. Navigate to: "%BOBJ INSTALL DIR%\SAP BusinessObjects Enterprise XI 4.0\win64_x64"
2. Run the command.

   Now, we can generate the certificate in two ways:
   - **Self-signed certificate** – CA and Server Certificates are generated using GENPSE and server certificate signing is also done using GENPSE.
   - **Generating CSR using GENPSE** – CA is generated using 3rd party library and server certificate csr using GENPSE after which, server certificate is signed by 3rd party CA using 3rd party tool (refer to section C).
3. To generate self-signed certificate, run the command: GenPSE.exe selfsigned temp.pse servercert.der cacert.der server.key passphrase.txt Default.cnf.

   Note: The .cnf file should be present in the win64_x64 location, which contains default values for the certificate generation like country name, state, and so on.
4. Enter the details. By default, it will take the values from the Default.cnf file.

You must follow the following rules while creating the default configuration file.

- You should add the values on the left-hand side exactly as mentioned below.
- The values on the left-hand side are case-sensitive.
- There should be only one space between a value and the 'equal to' (=) sign. For example, there is only one space between CA_Common_Name and the 'equal to' sign.
- You must ensure there is no space after the values on the right-hand side.

5. Follow the steps below to create a default configuration file:
   a) Open a new document in a text editor.
   b) Add the values as given below:

      CA_Common_Name = rootnm

      CA_Country = DE

      CA_State = BW

      CA_Locality = RRR

      CA_Email = root@gmail.com

      CA_Unit = root_u

      CA_Expiration[YYMMDD] = yymmdd

User_Expiration[YYMMDD] = yymmdd

User_Country = IN

User_State = KA

User_Locality = BLR

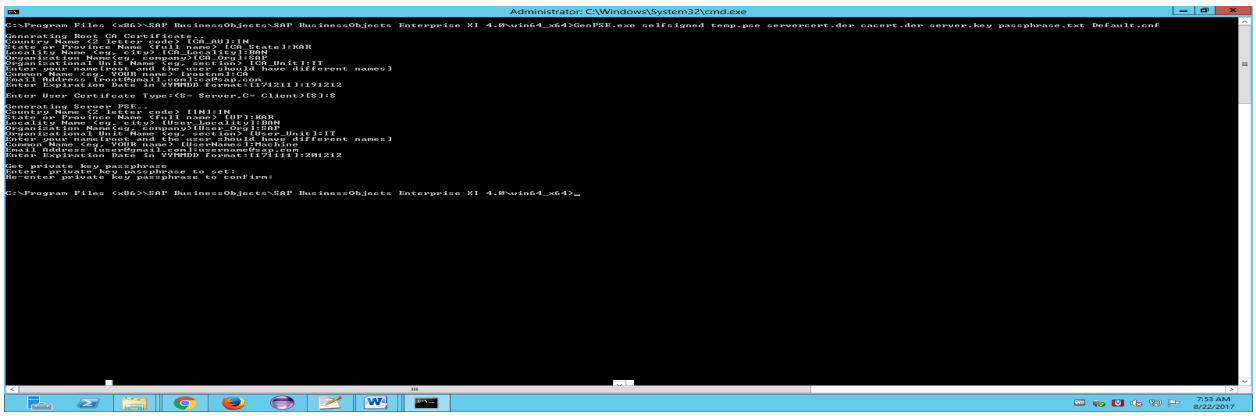
User_Organization = SSS

User_Unit = Unit

User_Common_Name = UserName

c) Save the file at <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64 with default.cnf name.

d) Make sure that Root CA certificate and PSE files are given different Common names.



After the above command is run, the following five files are created.

cacert.der

servercert.der

server.key

passphrase.txt

temp.pse

6. Place the above files in C:\SSL

   COPY cacert.der C:\SSL

   COPY servercert.der C:\SSL

   COPY server.key C:\SSL

   COPY temp.pse C:\SSL

   COPY passphrase.txt C:\SSL

4.3.3 Configure Tomcat to Communicate with a User's Browser Over HTTPS

1. Open Central Configuration Manager (CCM).
2. Stop Tomcat.
3. Navigate to server.xml path (%BOBJ INSTALL DIR%\tomcat\conf), keep a copy of server.xml.
4. Edit the server.xml file and search tag with port 8080. Add the below statement after the 8080 port tag.

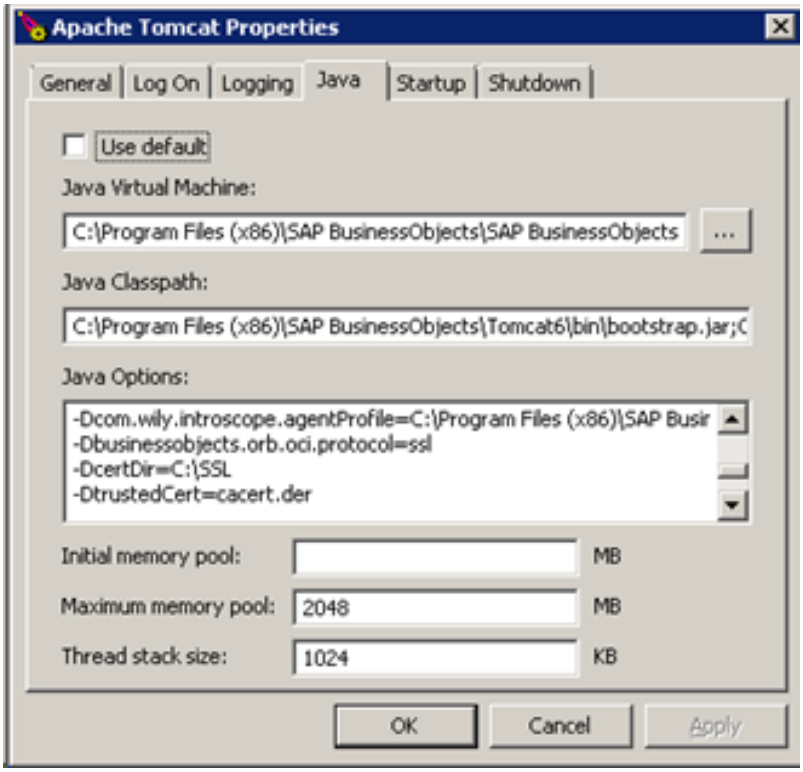
   <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true" maxThreads="150" scheme="https" secure="true" clientAuth="false" sslProtocol="TLS" minSpareThreads="25"

maxSpareThreads="75" enableLookups="false" disableUploadTimeout="true" acceptCount="100" debug="0" keystorePass="Password1" keystoreFile="C:\SSL\.keystore"/>

5. Save and close the server.xml file.

4.3.4 Configure Tomcat to Use the SSL Certificates for Communication with the SIA

1. Open Tomcat configuration
2. Go to the Java tab.



3. Add the text given below in the Java Options field.

   -Dbusinessobjects.orb.oci.protocol=ssl

   -DcertDir=C:\SSL

   -DtrustedCert=cacert.der

   -DsslCert=servercert.der

   -DsslKey=server.key

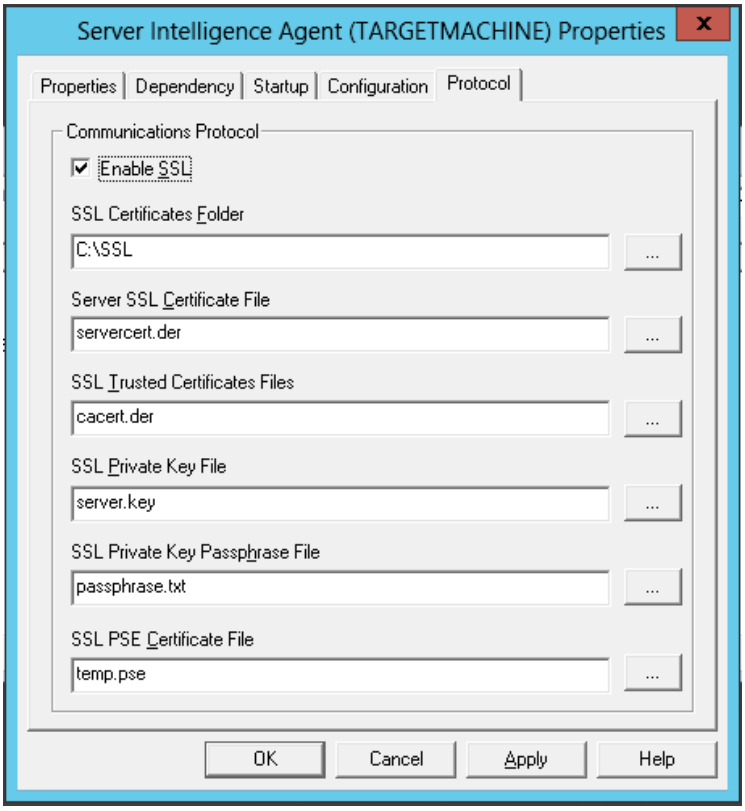
   -Dpassphrase=passphrase.txt

   Note: Do not include a space at end or beginning, otherwise, Tomcat won't start.

4. Click OK to start Tomcat again.

4.3.5 Configure the SIA to Use the SSL Certificates

1. In the CCM, stop the Server Intelligence Agent.
2. Double-click SIA and go to the Protocol tab.
3. Select Enable SSL.
4. Browse all files.



5. Click OK to start SIA.

   It should now be accessible using https://Servername(localhost):8443/BOE/CMC.

6. For setting SSL parameters, run the command:

   sslconfig.exe -dir C:/SSL -mycert servercert.der -rootcert cacert.der -mykey server.key -passphrase passphrase.txt -psecert temp.pse -protocol ssl

4.4 Create Users in BOE

Now that we have the list with details of the users from the IdP, we need to create/import them as Enterprise users in BOE from this SAP Cloud Platform Identity provider.

Users can be imported into BOE through the CSV file or by using an SDK script.

Note: The SAML-based authentication relies on Trusted Authentication from the Web server to the CMS. For this, the IdP users will have to be created in BOE as Enterprise users.

Please refer to the following link for importing bulk users from CMC:

https://blogs.sap.com/2013/05/16/bi-40-sp6-how-to-import-users-in-bulk-from-central-management-console/

4.5 Edit the securitycontext.xml file to Enable End Points

The securityContext.xml is located at <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF. In the securityContext.xml file, locate the SAML entry point in the XML code as below.

Please see the section below:

```
<security:http entry-point-ref="samlEntryPoint" use-expressions="false">
<!– Comment/Uncomment for Launchpad–>
<security:intercept-url pattern="/BI" access="IS_AUTHENTICATED_FULLY"/>
<!– Uncomment for Opendocument–>
<!–<security:intercept-url pattern="/OpenDocument/**" access="IS_AUTHENTICATED_FULLY"/>–>
<!– Uncomment for Fiori Launchpad–>
<!–<security:intercept-url pattern="/BILaunchpad" access="IS_AUTHENTICATED_FULLY"/>–>
<security:custom-filter before="FIRST" ref="metadataGeneratorFilter"/>
<security:custom-filter after="BASIC_AUTH_FILTER" ref="samlFilter"/>
</security:http>
```

Examples to configure for BI Platform Web applications:

- For BI Launchpad, by keeping this line uncommented <security:intercept-url pattern="/BI" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

- For OpenDocument, by keeping this line uncommented <security:intercept-url pattern="/OpenDocument/**" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

- For Fiorified BI Launchpad, by keeping this line uncommented <security:intercept-url pattern="/BILaunchpad" access="IS_AUTHENTICATED_FULLY"/>, under SAML entry point

Note: The XML tag for Classical BI Launchpad is enabled by default.

4.6 Changes in config Properties for BI Platform Web Applications

A new property "saml.enabled =true" has to be added in the respective BI Platform Web application config files.

As in any other properties' setting, it is recommended to put this property in the /config/custom/<application>.properties file.

If you do not already have any custom property file here, please create an empty <application>.property. To be sure, refer to the exact name in the /config/default directory.

Example

(Assuming custom properties file does not exist. If it already does, append the property **saml.enabled=true**)

- For Classic BI Launchpad, create BIlaunchpad.properties under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

- For Fiorified BI Launchpad, create fioriBI.properties under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

- For OpenDocument, create OpenDocument.properties under <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\config\custom

Add **saml.enabled =true**.

Note: It is mandatory to uncomment the specific endpoint and also add saml.enabled=true properties in custom properties file for the respective webapp, to enable SAML Authentication.

4.7 Configurations in the Deployment Descriptor— web.xml

A new filter has been introduced for SAML 2.0. The relevant section in the web.xml will be kept commented by default. Enable filters in web.xml of BOE webapp by uncommenting the SAML section(s).

Web.xml file path: <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF web.xml.

Uncomment the sections that have SAML comment as shown in the following images.

1. Uncomment the listener and context param.

   Commented listener and context param

```
        <!--SAML-->
    <!--<context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>
            /WEB-INF/securityContext.xml
        </param-value>
    </context-param>-->
    <!--SAML-->
    <!--SAML-->
    <!--<listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>-->

    <!--SAML-->
```

   After uncommenting the listener and context param the web.xml file looks as follows.

```
    <!--SAML-->
<context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>
        /WEB-INF/securityContext.xml
    </param-value>
</context-param>
<!--SAML-->
<!--SAML-->
<listener>
<listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>
```

2. Uncomment the SAML filters and mapping.

   Commented SAML filters and mapping

```
    <!--SAMl-->
<!--<filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
    </filter>

<filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
        <dispatcher>REQUEST</dispatcher>
        <dispatcher>FORWARD</dispatcher>
    </filter-mapping>-->
    <!--SAML ends -->
```

After uncommenting the SAML filters and mapping

```
    <!--SAMl-->
<filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
    </filter>

<filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
        <dispatcher>REQUEST</dispatcher>
        <dispatcher>FORWARD</dispatcher>
    </filter-mapping>
    <!--SAML ends -->
```

3. Save the web.xml with these changes.

4.8 Download SAML 2.0 IdP Metadata and Update in BOE

4.8.1 Download the SAML 2.0 IdP metadata from ADFS Identity provider

Launch the below URL in the machine containing ADFS for downloading IdP metadata file.

Note: The URL is of the form: https://adfs.adatum.com/federationmetadata/2007-06/federationmetadata.xml

4.8.2 Upload the SAML 2.0 IdP Metadata File to BOE

Copy the ADFS downloaded metadata file to <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF and rename it to idp-meta-downloaded.xml.

Note: By default, “idp-meta-downloaded.xml” name is generated in the securityContext.xml. We have to maintain the same name for the IdP metadata xml.

If BOE is deployed on a non-Windows platform, the path separators in the file path to the IdP metadata under the bean FilesystemMetadataProvider should be changed in securityContext.xml (<BOE Install Dir>\tomcat\webapps\BOE\WEB-INF).

That is, <value type="java.io.File">/WEB-INF/idp-meta-downloaded.xml</value> has to be changed to <value type="java.io.File">\WEB-INF\idp-meta-downloaded.xml</value> for Linux.

```
                <constructor-arg>
<bean class="org.opensaml.saml2.metadata.provider.FilesystemMetadataProvider">
                <!-- URL containing the metadata -->
                <constructor-arg>
                    <!--  <value type="java.io.File">file:///C:/idp-meta-downloaded.xml</value>-->
                    <value type="java.io.File">\WEB-INF\idp-meta-downloaded.xml</value>
                </constructor-arg>

                <property name="parserPool" ref="parse
            </bean>
                </constructor-arg>
                <constructor-arg>
                    <bean class="org.springframework.security.saml.metadata.ExtendedMetadata">
                    </bean>
                </constructor-arg>
                <property name="metadataTrustCheck"  value="false"/>
            </bean>
            <!-- Example of HTTP metadata without Extended Metadata -->
```

Path seperators changed to /WEB-INF/idp-meta-downloaded.xml (for linux) from \WEB-INF\idp-meta-downloaded.xml (forwindows)

4.9 Generate Keystore

SAML exchanges involve usage of cryptography for the signing and encryption of data. A sample self-signed keystore sampletestKeystore.jks is packaged with the product and is valid until October 18, 2019. The sampletestKeystore.jks file has an alias name Testkey and password Password1. You can now generate a self-signed keystore file using the JAVA utility keytool.

Follow the steps below to generate a keystore file:

1. Navigate to <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\bin
2. Run the command: keytool -genkey -alias <aliasname> -keypass <Password> -keystore < sampletestKeystore.jks> -keyalg RSA -validity <numberofdays>

Example

keytool -genkeypair -alias Testkey -keypass Password1 -keystore sampletestKeystore.jks -keyalg RSA -validity 735

Command	Description
-alias	Enter the alias name of the certificate
-keypass	Enter the certificate's password
-keystore	Name of the keystore file
-validity	Validity of the certificate
numberofdays	Number of days for which the self-signed certificate is valid

The following questions are prompted after executing the command:

- Enter keystore password: *****(Password1)
- Re-enter new password: *****(Password1)
- What is your first and last name?: <Name>
- What is the name of your organizational unit?: BusinessObjects
- What is the name of your organization?: SAP
- What is the name of your city and locality?: <CityName>
- What is the name of your State and Province?: <ProvinceName>

- What is the two-letter country code for this unit?: <CountryCode>

3. Stop the Tomcat Application Server.

   The keystore file is generated at <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\bin.

4. Move the keystore file to <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF.
5. Edit the xml file located at <INSTALLDIR>\tomcat\webapps\BOE\WEB-INF with the new alias name, password, and keystore file name.
6. Refer to the XML code below:

   <bean id="keyManager" class="org.springframework.security.saml.key.JKSKeyManager"><constructor-arg value="/WEB-INF/sampleKeystore.jks"/><constructor-arg type="java.lang.String" value="Password1″/><constructor-arg><map><entry key=" TestAlias " value="AliasPassword"/></map></constructor-arg><constructor-arg type="java.lang.String" value=" TestAlias "/></bean>

Refer to the table below for understanding the arguments.

Tag	Description
<constructor-arg value="/WEB-INF/sampleKeystore.jks"/>	Locates the keystore file
<constructor-arg type="java.lang.String" value="Password1″/>	Password for the keystore file
<entry key=" TestAlias " value=" AliasPassword"/>	Alias password
<constructor-arg type="java.lang.String" value=" TestAlias "/>	Alias of the default certificate

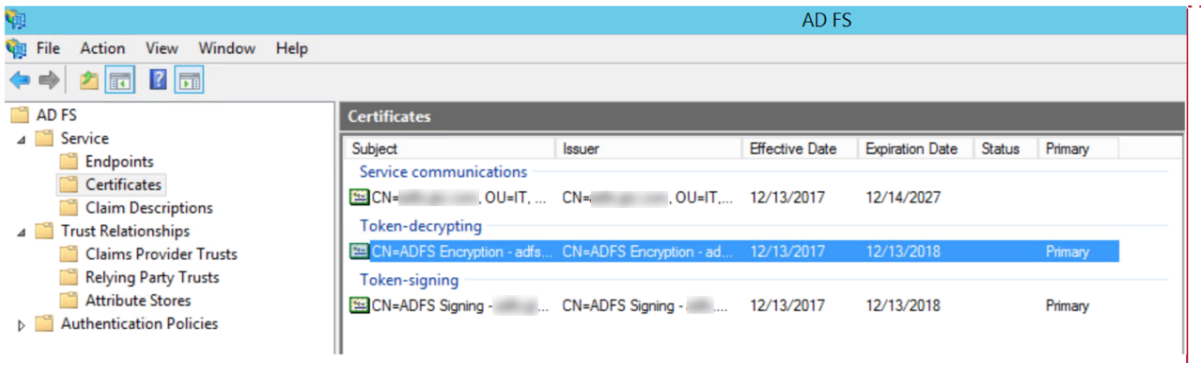
Note: SP metadata has to be generated every time this keystore file is changed. Our sample sp metadata will work only with our sample keystore certificate.

7. Restart the Tomcat Application Server.

4.10 Export the ADFS Certificates

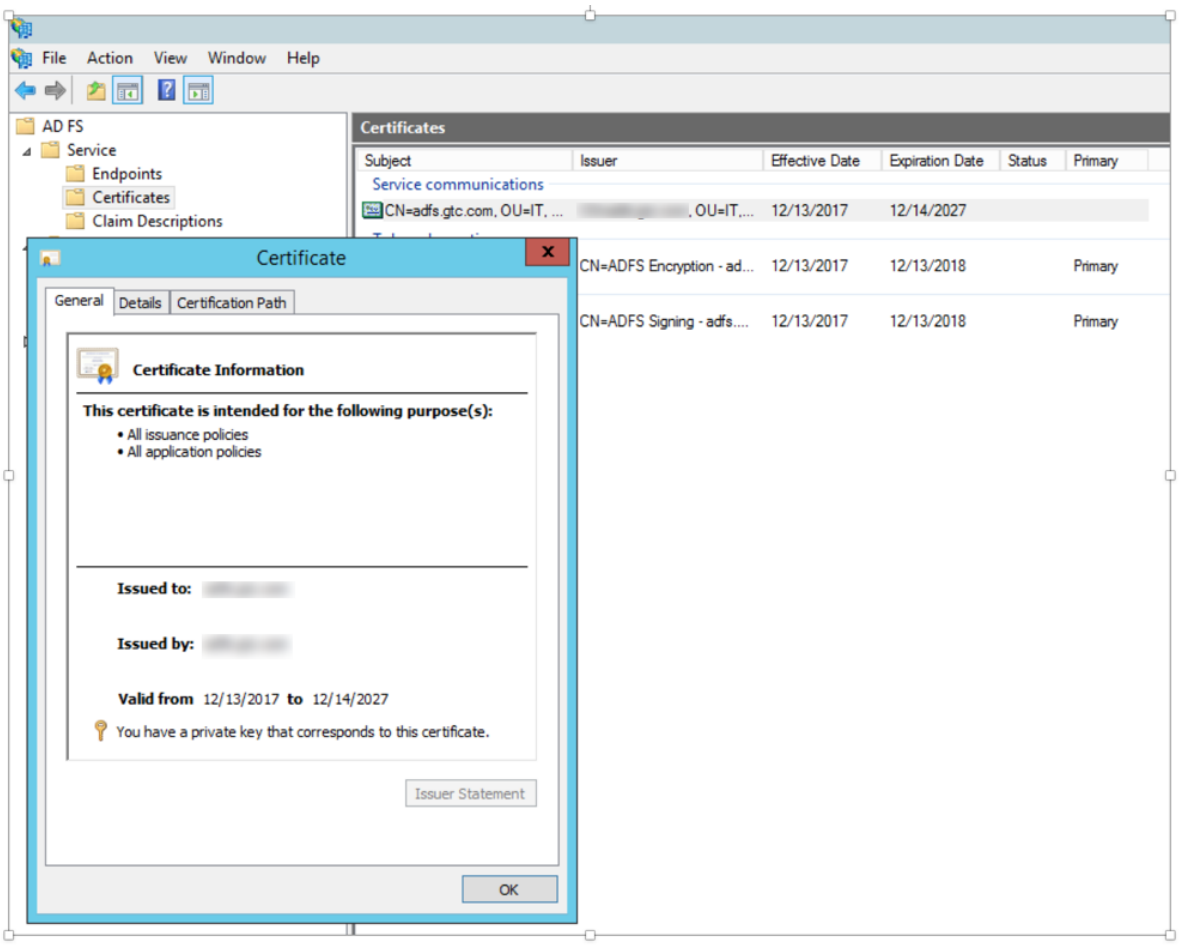
Download the certificates from ADFS server and transfer them to the Service Provider server.

1. Log on to the ADFS server.
2. Find the certificates on the "ADFS Management".

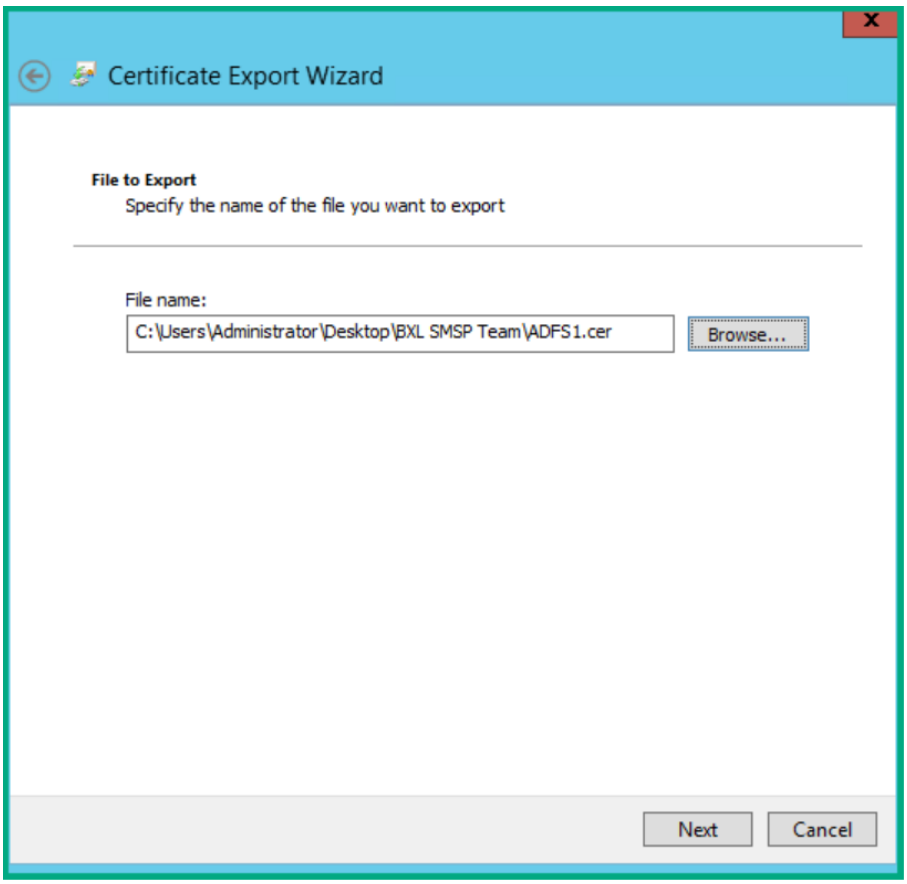


There should be three certificates – one for service communications, one for token-decrypting, and one for token signing.

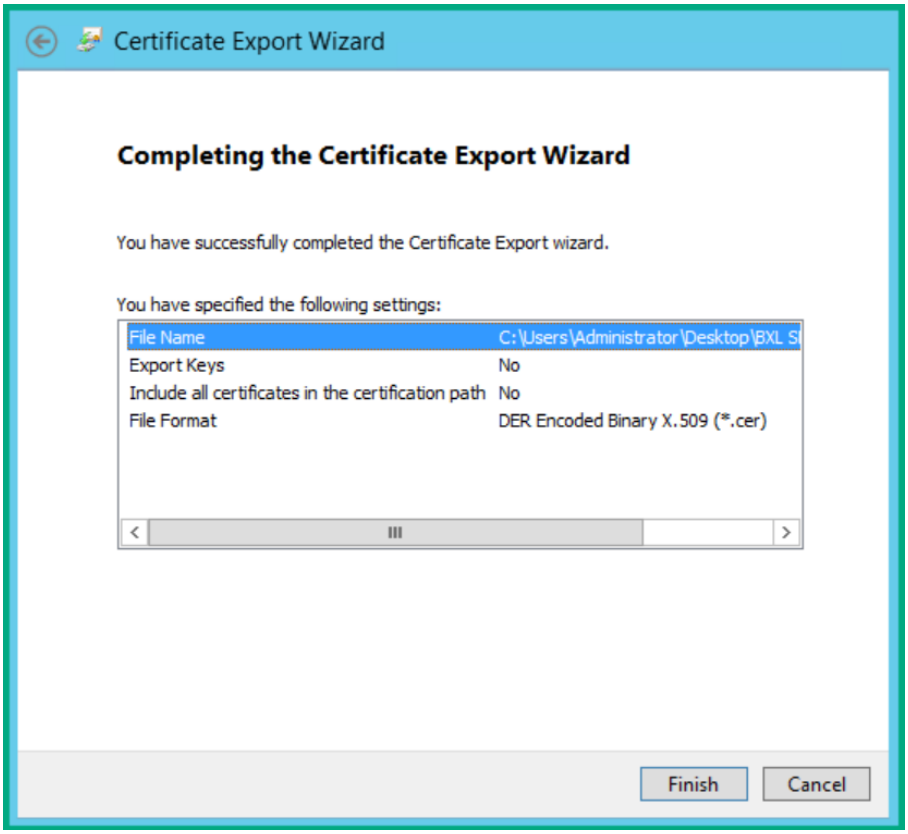
3. Right-click the first certificate, select View Certificate, go to the Details tab, and then click the Copy the File button.



4. Specify the export file name, and then click Next.



5. Click Finish to finish the export.



6. Repeat the previous steps for the other two certificates.

Name	Date modified	Type	Size
ADFS1	1/12/2017 10:31 PM	Security Certificate	2 KB
ADFS2	1/12/2017 10:32 PM	Security Certificate	1 KB
ADFS3	1/12/2017 10:32 PM	Security Certificate	1 KB

4.11 Import the ADFS Certificates Into the SP SAML Keystore

Import the three certificates to the SP SAML keystore located in <BOE Install Dir>\tomcat\webapps\BOE\WEB-INF\sampletestKeystore.jks.

Run the following command for each of the three certificates.

keytool -v -importcert -file <certificate filename> -keystore sampletestKeystore.jks -alias <certificate alias>

Example

keytool -v -importcert -file ADFS1.cer -keystore sampletestKeystore.jks -alias Test1

4.12 Generate and Upload SP Metadata

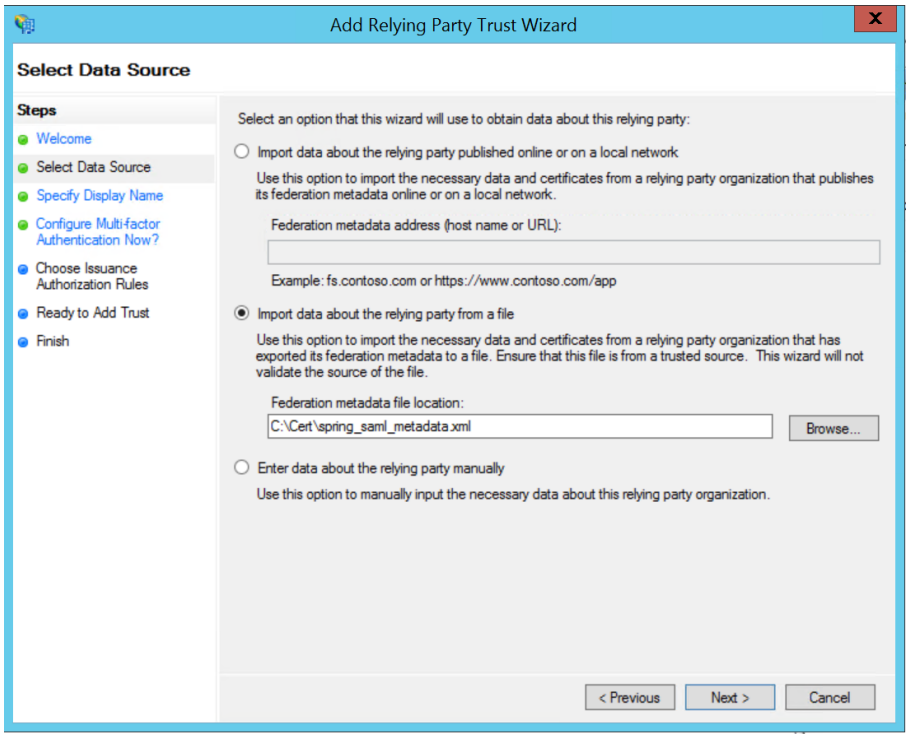
Note: A pre-generated Service Provider (SP) metadata file will be shipped by default. User may edit this and upload the same. The IP/hostname should be one property that has to be changed. The file will be available under: <BOE Install Dir>\tomcat\webapps\biprws\WEB-INF spring_saml_metadata.xml

Type the URL https://BOEHOST:8443/BOE/saml/metadata.

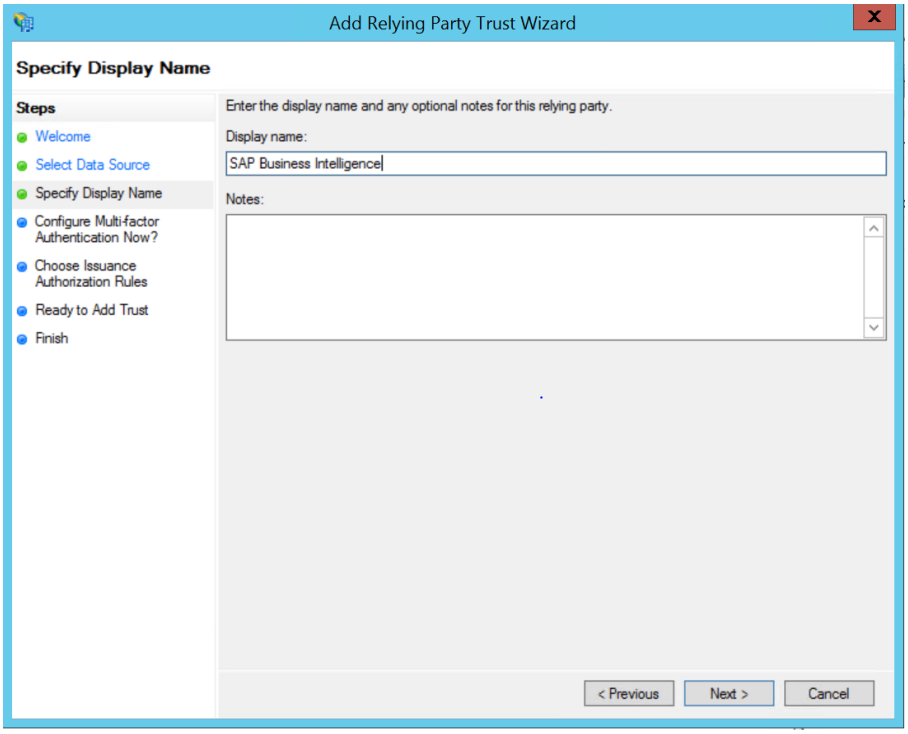
This will automatically download an xml file spring_saml_metadata.xml.

4.13 Import the Service Provider Metadata file in ADFS

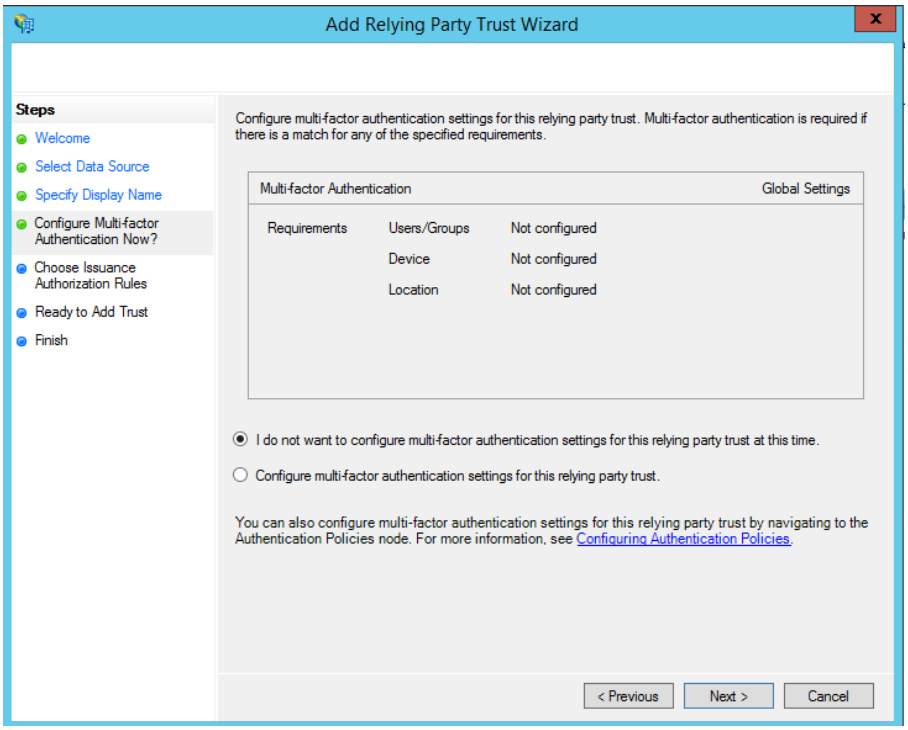
1. Add Relying Party Trust — Import the Service Provider metadata file in ADFS.



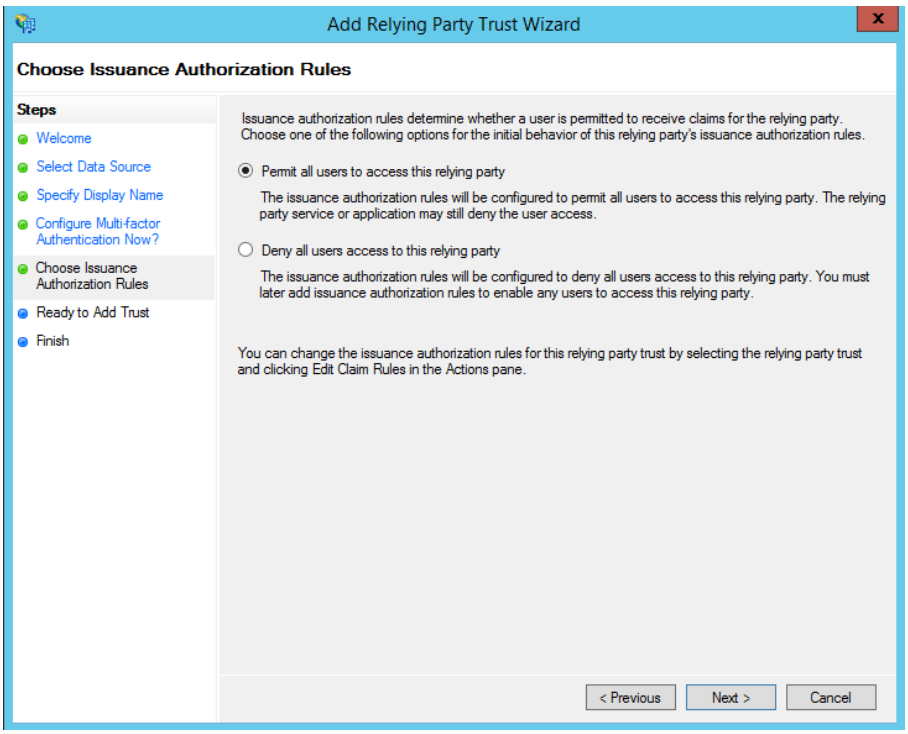
2. After importing the file, click Next.
3. Specify Display name and click Next.



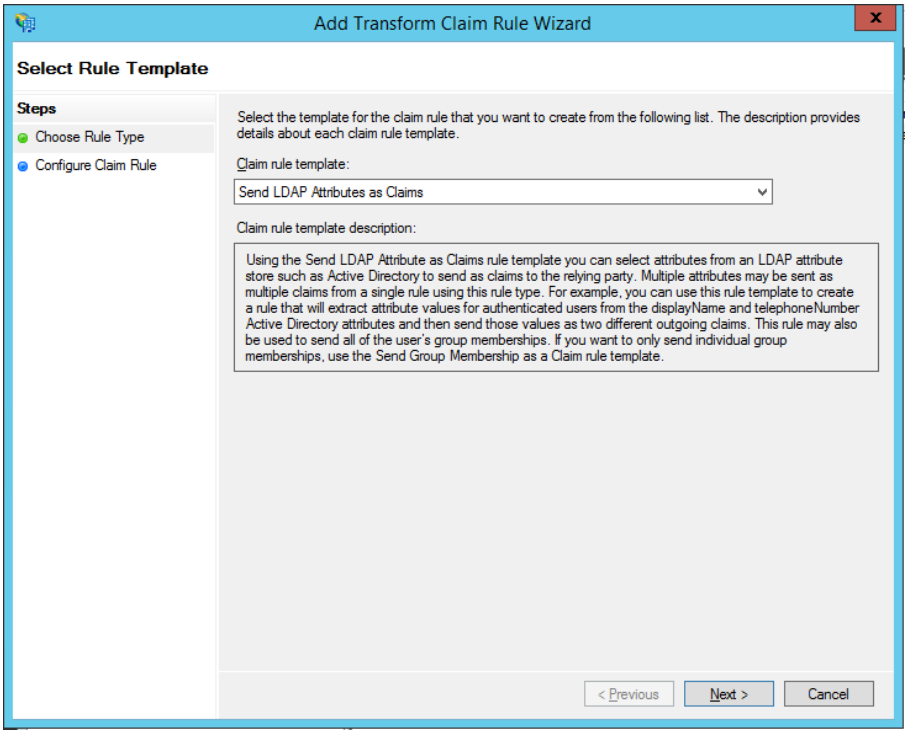
4. Select I do not want to configure multi-factor authentication settings for this relying party trust at this time.



5. Under Issuance Authorization Rules, select Permit all users to access this relying party.

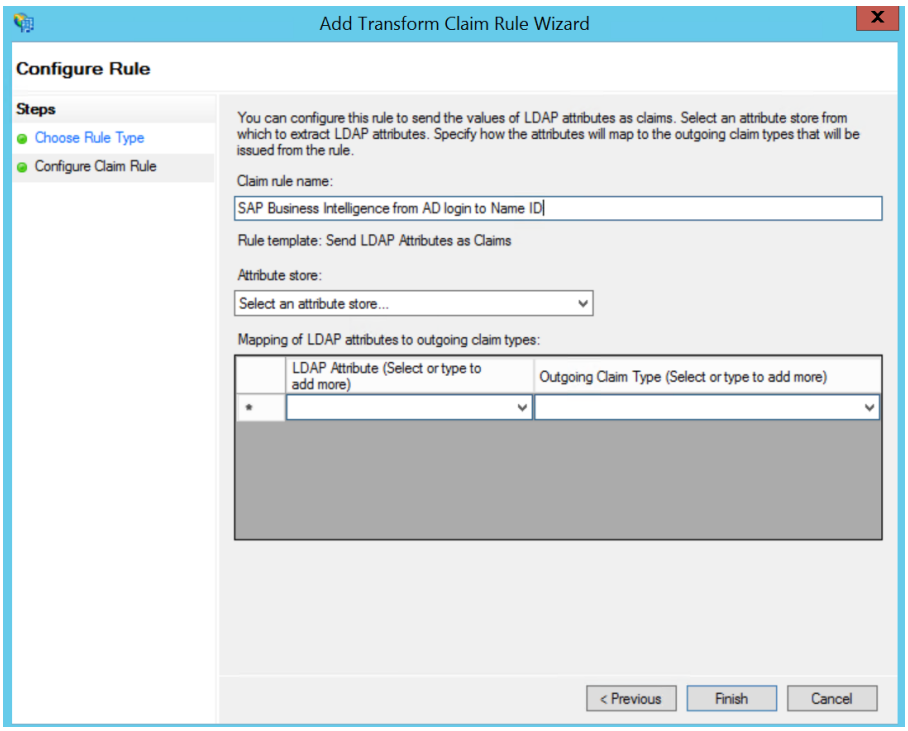


6. Click Next and then click Finish.
7. Add Claim Rule for SAP Business Intelligence.
8. Select Send LDAP Attribute as Claims and click Next.

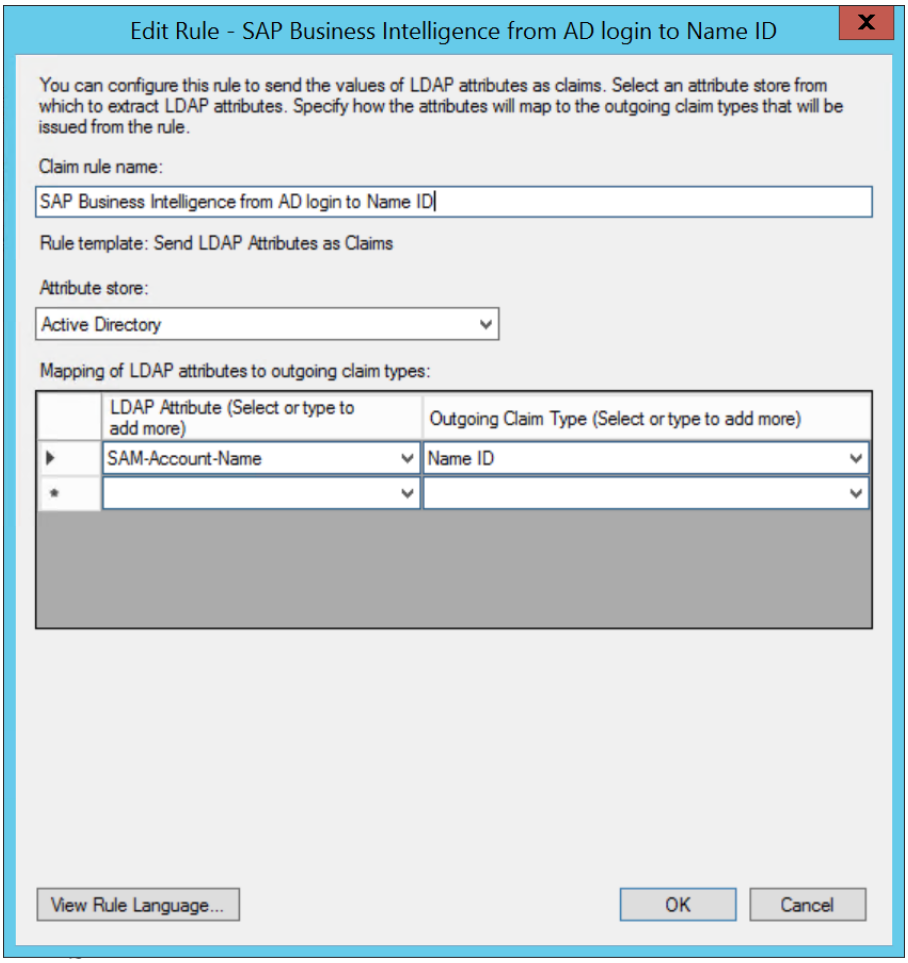


9. Enter Claim Rule name.

   SAP Business Intelligence from AD login to Name ID.

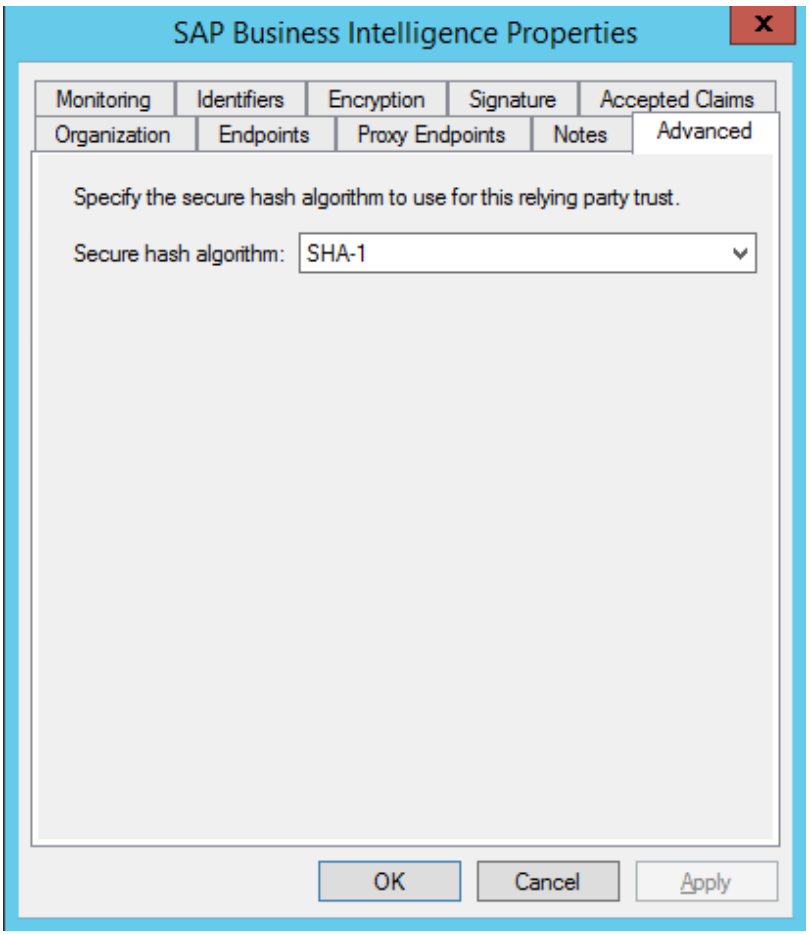


10. Select attribute store — Active Directory and mapping of LDAP attributes.



This is a transformation example, from Login name in active directory to Name ID that can be used in SAP Business Intelligence.

11. Go to SAP Business Intelligence replying party properties → Advanced, and change the secure hash algorithm to SHA-1.



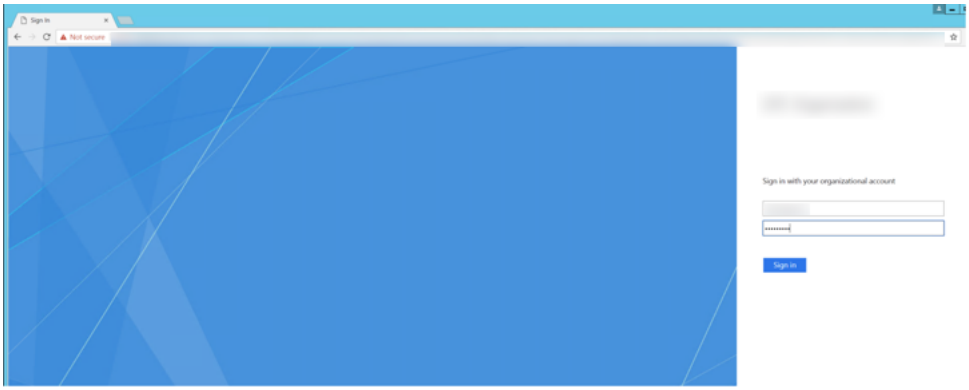
Verification

SAML is configured for BI Launchpad endpoint

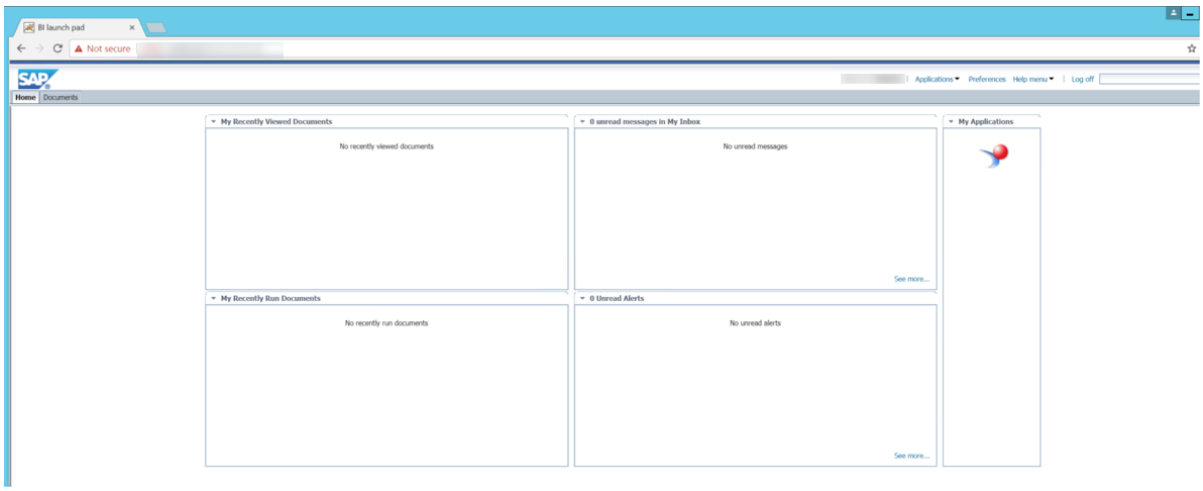
1. Hit the URL https://BOEHOST:8443/BOE/BI.

   It redirects to IdP authentication.

2. Enter your domain users' details.



If the configuration is correct and mapping is successful and trusted authentication is configured correctly, you will be logged on to BOE/BI.



This completes the configuration.

Users will be able to use SAML to log in to SAP Business Intelligence.

Disclaimer

If ADFS certificate keys are not exchanged properly with BOE certificate, you may face the issue while redirecting the URLs.

To resolve this issue, follow the steps given below.

1. Go to added Relying party → Properties → Encryption and remove the certificate.
2. Make sure the certificate is added to the Signature tab.

This is required for SAML token sign.

5. ISSUES AND CHALLENGES

Helpful Tools

- HttpWatch or Fiddler to capture HTTP traffic; SAML metadata is captured in these tools
- Base64 Decoder (example); SAML metadata is encoded in Base64
- URL Decoder (example); Some tools capture SAML Metadata with the URL encoded (%20 instead of space)
- XML editor, such as Notepad++; Makes editing and viewing XML files a bit easier over traditional Notepad

Expected Workflow

Reference : https://launchpad.support.sap.com/#/notes/2604208

Once the configuration is complete, the following workflow is expected (assuming no previous authentications are cached):

1. User accesses the configured front-end (BI Launchpad, OpenDocument, Fiori BI Launchpad, etc.).

Result	Protocol	Host	URL
200	HTTP	fp-425-sso:8080	/BOE/BI ← Start
200	HTTP	Tunnel to	psauth-sci.accounts400.ondemand.com:443

2. The Tomcat Web application automatically routes to the IdP logon page.tomcat Web application and routes to the IdP logon page.

Result	Protocol	Host	URL
200	HTTP	fp-425-sso:8080	/BOE/BI
200	HTTP	Tunnel to	psauth-sci.accounts400.ondemand.com:443
200	HTTPS	psauth-sci.acc...	/saml2/idp/sso/psauth-sci.accounts400.ondemand.com

3. The user provides the IdP logon information.
4. The IdP authenticates the user and re-routes the user back to the configured SAML endpoint on Tomcat.

Result	Protocol	Host	URL
200	HTTPS	psauth-sci.acc...	/saml2/idp/sso/psauth-sci.accounts400.ondemand.com
302	HTTP	fp-425-sso:8080	/BOE/saml/SSO

5. Tomcat redirects to the original starting point (BI Launchpad, OpenDocument, Fiori BI Launchpad, etc.).

Result	Protocol	Host	URL
302	HTTP	fp-425-sso:8080	/BOE/saml/SSO
200	HTTP	fp-425-sso:8080	/BOE/BI

Note the HTTP 302 result indicating a successful response. If BOE/saml/SSO hit ends in an HTTP 404 result, something has gone wrong with the previous step. There is no landing or accessible page on /BOE/saml/SSO URL. Accessing this URL directly will always cause 404 response.

6. At this stage, Tomcat BOE Web Application logs show the start of communication for the authentication attempt.

```
LogonComponentRenderer processSSO : END -- EntSession is not NULL And AppKind is :  InfoView
LogonComponentRenderer processSSO : RedirectToSuccess And AppKind is :  InfoView
Begin writing state to response for viewId/logon.jsp
End writing state to response for viewId/logon.jsp
```

Note: Text from log shows LogonComponentRenderer processSSO

7. Tomcat uses SAML metadata to perform Trusted Authentication and log in to CMS.

Result	Protocol	Host	URL
200	HTTP	fp-425-sso:8080	/BOE/BI
200	HTTP	fp-425-sso:8080	/BOE/portal/1801021405/InfoView/logon.faces

6. COMMON PROBLEMS

6.1. Presented with Logon Screens After Successful SAML Authentication

Symptom

- Configured SAML on Tomcat
- Launch BI Application like BI Launchpad or Fiori Launchpad
- Presented with IdP Logon Screen, enter credentials
- Presented with Login Screen of BI Application

Cause

- Trusted Authentication Configuration would be missing or user coming from IdP does not exist as a user in Enterprise User in BOE.

Solution

Perform the following actions:

1. Configure Trusted Authentication with WebSession
   - Add the global.properties file under the custom folder <INSTALLDIR>\SAPBusinessObjects\tomcat\webapps\BOE\WEB-INF\config\custom.

In case global.properties file exists under custom folder, the trusted authentication configuration has to be appended to the existing file.

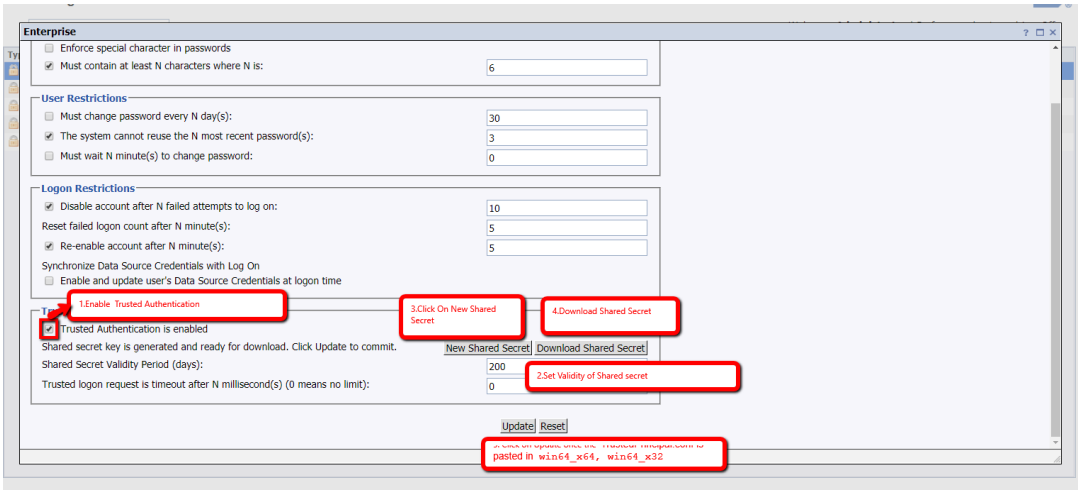
Following is the content for global.properties:

sso.enabled=true

trusted.auth.user.retrieval=WEB_SESSION

trusted.auth.user.param=UserName

2. Configure Trusted Auth in CMC.
   - Go to CMC Application, Authentication , Enterprise . Refer to the screen below.



3. Enable Trusted Authentication.
4. Set the Validity.
5. Choose New Shared Secret.
6. To download the generated shared secret, choose *Download Shared Secret.*

   The TrustedPrincipal.conf file is downloaded.
7. Paste the TrustedPrincipal.conf file in <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x64and <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\win64_x32
8. Go to CMC Authentication Enterprise, and choose Update.
9. Restart Tomcat.

6.2. User Creation on BOE

The IdP user has to be created in BOE or imported through some SDK script or exported using CSV option in CMC. The SAML based authentication relies on TrustedAuth from the WebServer to the CMS. For this, the IdP users will have to be created in BOE as Enterprise users.

6.3. SAML Authentication Fails When Tomcat is Behind Load Balancer or Reverse Proxy

Reference
https://launchpad.support.sap.com/#/notes/2621904

When configuring SAML on the Web application in a clustered mode behind a load balancer, the back-end nodes need to be instructed about the public URL on the front end. Additional configuration is required.

Symptom

- Configure SAML Authentication
- SAML workflow fails when using front-end URL

Cause

When configuring SAML on the Web application in a clustered mode behind a load balancer, the back-end nodes need to be instructed about the public URL on the front end. Additional configuration is required.

Solution

Make sure that your reverse-proxy or load-balancer is configured to use sticky sessions.

1. Open securityContext.xml in an editor.
2. Comment out the configured "contextProvider" bean:

Original	Modified
<bean id="contextProvider" class="org.springframework.security.saml.context.SAMLContextProviderImpl"/>	<!--<bean id="contextProvider" class="org.springframework.security.saml.context.SAMLContextProviderImpl"/>-->

3. Add a new "contextProvider" bean and update with appropriate values.

Load Balancer contextProvider Bean
<bean id="contextProvider" class="org.springframework.security.saml.context.SAMLContextProviderLB"> <property name="scheme" value="https"/> <property name="serverName" value="www.example.com"/> <property name="serverPort" value="443"/> <property name="includeServerPortInRequestURL" value="false"/> <property name="contextPath" value="/BOE"/> </bean>

4. Restart Tomcat Web Application Server.
5. Repeat the above steps for each Tomcat node in the cluster.
6. Generate the Service Provider metadata using the front-end URL (Go to https://www.example.com:443/BOE/saml/metadata).
7. Upload the generated metadata file to the Identity provider (IdP).

More information about load balancer and reverse proxy configuration for the SAML extension can be found at the Spring SAML Security extension reference site: https://docs.spring.io/autorepo/docs/spring-security-saml/1.0.x/reference/html/configuration-advanced.html.

6.4. How to Enable Trace Logging for BI SAML Extension (log4j)

Reference
https://launchpad.support.sap.com/#/notes/263442

Symptom

- Need to collect debugging information from the BI SAML extension used for front-end authentication.
- Troubleshooting SAML workflow between Identity provider and SAML extension.
- BOE Web application logs do not contain information regarding SAML workflow.
- Tomcat standard output logs do not contain information regarding SAML workflow.

Reproducing the Issue

1. Configure SAML Authentication for BOE on Tomcat.
2. SAML Authentication fails.

Solution

To enable tracing of the Spring Security SAML extension included with BI 4.2 SP05+, the following files need to be modified:

1. Back up the web.xml file and the securityContext.xml file found in the <Tomcat>\webapps\BOE\WEB-INF\ directory.
2. Modify the web.xml file by adding the following section, and then save.

web.xml file
<!-- logging --> <context-param> <param-name>log4jConfigLocation</param-name> <param-value>/WEB-INF/log4j.properties</param-value> </context-param> <listener> <listener-class>org.springframework.web.util.Log4jConfigListener</listener-class> </listener> <!-- logging -->

Note: When adding the tags to the web.xml file, add them within the <web-app> </web-app> tags.

3. Modify the securityContext.xml file by making the following changes, and then save the file.

Original	Modified
<!-- Logger for SAML messages and events --> <bean id="samlLogger" class="org.springframework.security.saml.log.SAMLDefaultLogger"/>	<!-- Logger for SAML messages and events --> <bean id="samlLogger" class="org.springframework.security.saml.log.SAMLDefaultLogger"> <property name="logMessages" value="true" /> <property name="logErrors" value="true" /> </bean>

4. Create a file called log4j.properties in the <Tomcat>\webapps\BOE\WEB-INF\ directory.

5. Add the following content to the log4j.properties file, and then save the file.

web.xml file

# Root logger option 	
log4j.rootLogger=DEBUG, file
# Redirect log messages to a log file 	
log4j.appender.file=org.apache.log4j.RollingFileAppender
#outputs to Tomcat home
log4j.appender.file.File=${catalina.home}/logs/springsaml.log
log4j.appender.file.MaxFileSize=5MB
log4j.appender.file.MaxBackupIndex=10
log4j.appender.file.layout=org.apache.log4j.PatternLayout

web.xml file

log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n
#Spring Security SAML Extension Debugging
log4j.logger.org.springframework.security.saml=DEBUG
log4j.logger.org.opensaml=DEBUG
log4j.logger.PROTOCOL_MESSAGE=DEBUG

6. Restart Tomcat.
7. After the changes have been made, debugging information will be logged in a new file called "springsaml.log" located in the <Tomcat>\logs\ directory.

7. USEFUL RESOURCES

7.1. SCN Blog Posts

Title and Author: SAML Authentication for BOE on Tomcat by Shruthi Annappa of SAP.

URL: https://blogs.sap.com/2017/11/17/saml-authentication-for-boe-on-tomcat

Abstract: In this blog post, Shruthi provides a high-level overview of SAML Authentication for BOE on Tomcat with SAP Cloud Platform Identity provider as an IdP and its related configuration as Part of the BI Platform 4.2 SP05.

Title and Author: SAML Authentication Rest Endpoint for BOE on Tomcat by Shruthi Annappa of SAP.

URL: https://blogs.sap.com/2017/11/16/saml-authentication-rest-endpoint-for-boe-on-tomcat

Abstract: In this blog post, Shruthi provides a high-level overview of SAML Authentication Rest Endpoint for BOE on Tomcat with SAP Cloud Platform Identity provider as an IdP and its related configuration as Part of the BI Platform 4.2 SP05.

Title and Author: ADFS with SAP Business Intelligence Platform by Dhrubajyoti Paul of SAP.

URL: https://blogs.sap.com/2018/02/22/adfs-with-sap-business-intelligence-platform/

Abstract: In this blog post, Dhrub provides a high-level overview of SAML Authentication for BOE on Tomcat with ADFS as an IdP and its related configuration as Part of the BI Platform 4.2 SP05.

Title and Author: Hybrid Authentication for SAP Analytics HUB – SAML SSO to BI Platform content by Ashok Rajashekar of SAP.

URL: https://blogs.sap.com/2017/12/19/sap-analytics-hub-saml-sso-to-BI Platform-content

Abstract: In this blog post, Ashok provides a high-level overview of Hybrid Authentication for SAP Analytics HUB – SAML SSO to BI Platform content as Part of the BI Platform 4.2 SP05.

7.2. SAP Notes

Here is the list of some important and relevant SAP Notes on SAML.

SAP Note Number: SAP Note 2604208.

URL: https://launchpad.support.sap.com/#/notes/2604208

Abstract: This SAP note focuses on BI Auth Troubleshooting Series: SAML Authentication on Tomcat's BOE Web Application.

SAP Note Number: SAP Note 1795949.

URL: https://launchpad.support.sap.com/#/notes/1795949

Abstract: This SAP note focuses on Trusted Authentication with SAML Single Sign-On BI 4.x.

7.3. Acronyms

Acronym	Definition
BOE	Business Objects Enterprise
IdP	Identity provider
SP	Service Provider
BI Platform	Business Intelligence Platform
Webapps	Web applications
ADFS	Active Directory Federation services
SSO	Single Sign-On
CMC	Central Management Console
PAM	Product Availability Matrix
SSL	Secure Sockets Layer
CMS	Central Management Server
SIA	Server Intelligence Agent
URL	Uniform Resource Locator