erwin **Data Intelligence (DI)**

Version 14.0

Single Sign-On (SSO) configuration Guide

This document provides the instructions to configure Single Sign-On (SSO) for erwin DI.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.

Attn: LEGAL Dept

4 Polaris Way Aliso Viejo, CA 92656

Refer to our Web site (https://www.quest.com) for regional and international office information.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at https://www.quest.com/legal.

Trademarks

Contents

About this Guide

This document is an administration guide that describes how to configure and manage Single Sign On (SSO) based on SAML 2.0 for erwin Data intelligence Suite.

Introduction

The identity federation standard Security Assertion Markup Language (SAML) 2.0 enables the secure exchange of user authentication data between erwin Data Intelligence Suite and identity providers.

When you use the SAML 2.0 protocol to enable single sign-on (SSO), security tokens containing assertions pass information about an end user (principal) between a SAML authority - an identity provider (IdP), and a SAML consumer - a service provider (SP). erwin Data Intelligence Suite, acts as the service provider (SP).

Pre-Requisites

- This configuration guide is developed considering that SAML is already configured within customer's organization, and erwin Data intelligence Suite is being added to it.
- Further, our assumption is that the customer has an internal SAML administrator.
- We would need SAML administrator assistance to generate idp.xml, once erwin provides service provider metadata to the SAML administrator.
- If customers want to use their own security certificate and private key files, then respective details need to be provided in the database.properties configuration file.

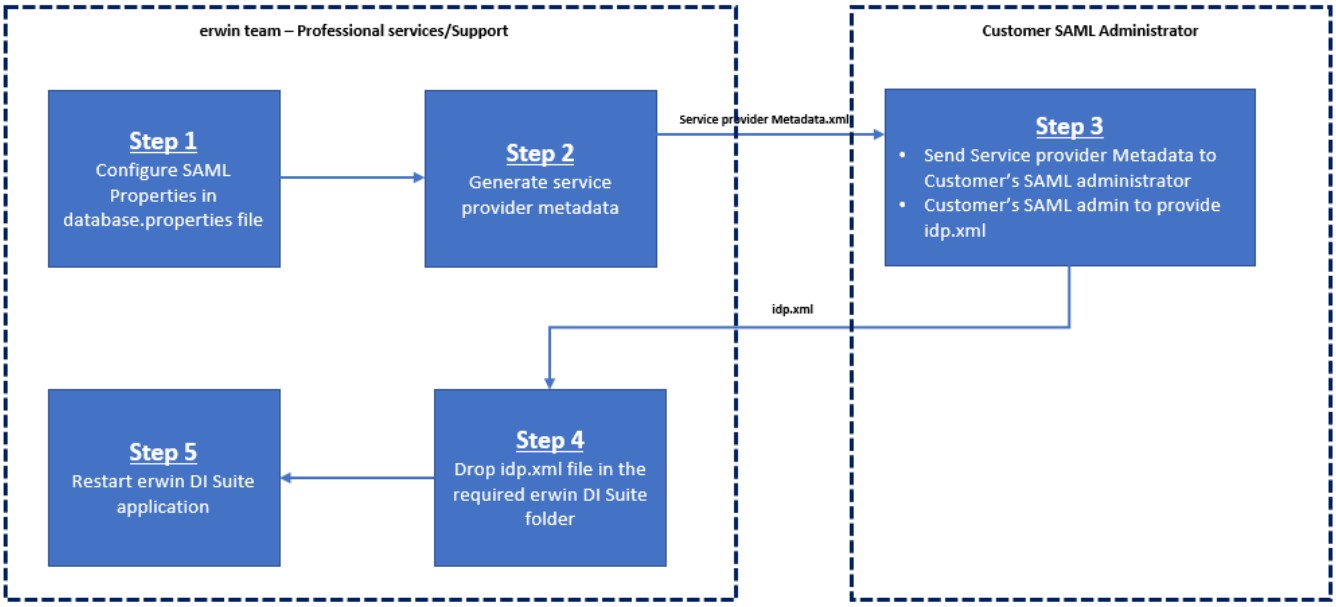
Supported SAML authentication providers

erwin Data Intelligence supports Single Sign-on (SSO) through SAML using external identity providers (IdPs) such as

1. Okta
2. OneLogin
3. SiteMinder
4. PingFederate
5. Microsoft Active Directory Federation Service.

Additionally, erwin Data Intelligence is compatible with other external IdPs that support SAML 2.0.

Configuration steps



Step 1: Configure SAML Properties in database.properties file

1. Navigate to database.properties file at the below location
   C:\Program Files\Apache Software Foundation\Tomcat\webapps\erwinDIsuite\WEB-INF\database\database.properties

2. Update WebAuthenticationMechanism from "DB" to "DB, SAML"

```
### Authentication Configuration Begin
WebAuthenticationMechanism=DB, SAML
WebServiceAuthenticationMechanism=DB
security.roles.authorization.saml.enabled=true
security.roles.authorization.ldap.enabled=true
### Authentication Configuration End
```

   Set **security.roles.authorization.saml.enabled= false** to support only user authentication against SAML and ignore fetching the roles/groups from SAML.
   Set **security.roles.authorization.saml.enabled= true** to support the older state of the application that includes both user authentication and fetching of role/groups from SAML.

3. Steps to update SAML configuration section parameters

   Step 3.1
   spring.security.saml2.customrelyingparty.**samlentityid**

   - If you are configuring SAML for erwin DI for the first time, please comment this parameter or delete the same [Sample configuration in reference image below]

- If SAML has already been configured on erwin DI previously then update this parameter to the same value of SAMLEntityId parameter present in your old database.properties file [Prior to erwin DI 13.2]

spring.security.saml2.customrelyingparty.samlentityid = SAML Entity ID (configured in old database.properties file)

Eg: spring.security.saml2.customrelyingparty.samlentityid= **erwinDev**

Step 3.2

spring.security.saml2.relyingparty.registration.{**registerID**}.signing.credentials[0].certificate-location=classpath:resources/**local**.crt

- Replace register id with your relevant SAML IdP provider name. Any unique name should be fine. [In the below example MSAD is used as register ID for Microsoft Active Directory]
- To utilize the client specific security certificate, copy the specific security certificate .crt file into the resources folder present in tomcat\webapps\erwinDISuite\WEB-INF\classes\resources folder
- Update local.crt parameter to the client specific security certificate file name copied in the previous step. [In the below example Quest.crt is the file name]

Example:
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].certificate-location=classpath:resources/Quest.crt

Step 3.3

*spring.security.saml2.relyingparty.registration.{**registerID**}.signing.credentials[0].private-key-location=classpath:resources/**local**.key*

- Replace register id with relevant SAML IdP provider name. Any unique name should be fine.
- To utilize the client specific private key file, copy the private .key file into the resources folder present in tomcat\webapps\erwinDISuite\WEB-INF\classes\resources folder.
- Update local.key parameter to the client specific private key file name copied in the previous step. [In the below example QSoft.key is the private key file name]

Example:
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].private-key-location=classpath:resources/QSoft.key

Note regarding *{registerID} in #2 and #3 above:*
In the erwin DI 13.2 configuration file, you will see *okta* instead of *registerID*. Replace *okta* with the appropriate *registerID* value provided by your SAML external identity provider.

Step 3.4

spring.security.saml2.customrelyingparty.algorithm=http://www.w3.org/2000/09/xmldsig#rsa-sha1

- Update the URL with the relevant security algorithm encryption required.
- List of supported algorithm URLs are below:
  - http://www.w3.org/2000/09/xmldsig#dsa-sha1
  - http://www.w3.org/2000/09/xmldsig#rsa-sha1
  - http://www.w3.org/2000/09/xmldsig#hmac-sha1
  - http://www.w3.org/2000/09/xmldsig#sha1
  - http://www.w3.org/2001/04/xmldsig-more#rsa-md5
  - http://www.w3.org/2001/04/xmldsig-more#rsa-ripemd160
  - http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
  - http://www.w3.org/2001/04/xmldsig-more#rsa-sha384
  - http://www.w3.org/2001/04/xmldsig-more#rsa-sha512
  - http://www.w3.org/2001/04/xmldsig-more#hmac-md5
  - http://www.w3.org/2001/04/xmldsig-more#hmac-ripemd160

- http://www.w3.org/2001/04/xmldsig-more#hmac-sha256
- http://www.w3.org/2001/04/xmldsig-more#hmac-sha384
- http://www.w3.org/2001/04/xmldsig-more#hmac-sha512
- http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha1
- http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256
- http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha384
- http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha512
- http://www.w3.org/2001/04/xmldsig-more#md5
- http://www.w3.org/2001/04/xmldsig-more#sha384
- http://www.w3.org/2001/04/xmlenc#sha256
- http://www.w3.org/2001/04/xmlenc#sha512
- http://www.w3.org/2001/04/xmlenc#ripemd160

Step 3.5
spring.security.saml2.customrelyingparty.samlrolesseparator = ^

- Caret ^ is provided as the default Role separator. Please update as required.

Example with role separator as Dollar symbol $
spring.security.saml2.customrelyingparty.samlrolesseparator =$

Step 3.6
spring.security.saml2.customrelyingparty.firstname=FirstName

- *FirstName* is provided as the default *firstname* attribute. Please update as per the attribute name configured in the respective SAML IdP provider by your SAML administrator.

Please note that the attribute name is case sensitive.

Step 3.7
spring.security.saml2.customrelyingparty.lastname=LastName

- *LastName* is provided as the default *lastname* attribute. Please update as per the attribute name configured in the respective SAML IdP provider by your SAML administrator.

Please note that the attribute name is case sensitive.

Step 3.8
spring.security.saml2.customrelyingparty.emailaddress=EmailAddress

- *EmailAddress* is provided as the default *emailaddress* attribute. Please update as per the attribute name configured in the respective SAML IdP provider by your SAML administrator.

Please note that the attribute name is case sensitive.

Step 3.9
spring.security.saml2.customrelyingparty.userrole=amm_user_role

- *amm_user_role* is provided as the default userrole attribute. Please update as per the attribute name configured in the respective SAML IdP provider by your SAML administrator.

Please note that the attribute name is case sensitive.

Sample SAML configuration for first time SAML erwin DI configuration

```
############################################################
#spring.security.saml2.customrelyingparty.samlentityid=springDevDIS #commented as SAML erwin DI integration is being done for first time
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].certificate-location=classpath:resources/Quest.crt
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].private-key-location=classpath:resources/QSoft.key
spring.security.saml2.customrelyingparty.algorithm= http://www.w3.org/2000/09/xmldsig#rsa-sha1
spring.security.saml2.customrelyingparty.samlrolesseparator=$
spring.security.saml2.customrelyingparty.firstname=FirstName
spring.security.saml2.customrelyingparty.lastname=LastName
spring.security.saml2.customrelyingparty.emailaddress=EmailAddress
spring.security.saml2.customrelyingparty.userrole=amm_user_role
#saml configuration ends
```

Sample SAML configuration for clients who have SAML configured on erwin DI prior to DI 13.2

```
############################################################
spring.security.saml2.customrelyingparty.samlentityid=ErwinDev
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].certificate-location=classpath:resources/Quest.crt
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].private-key-location=classpath:resources/QSoft.key
spring.security.saml2.customrelyingparty.algorithm= http://www.w3.org/2000/09/xmldsig#rsa-sha1
spring.security.saml2.customrelyingparty.samlrolesseparator=$
spring.security.saml2.customrelyingparty.firstname=FirstName
spring.security.saml2.customrelyingparty.lastname=LastName
spring.security.saml2.customrelyingparty.emailaddress=EmailAddress
spring.security.saml2.customrelyingparty.userrole=amm_user_role
#saml configuration ends
```

Step 3.6
Restart the erwinDI web application from tomcat console.

Step 2: Generate Service Provider Metadata.xml

1. If you are configuring SAML login for erwin DI for the first time, then:

   Open the browser and navigate to below URL to download SAML Service Provider Metadata.xml

   https://<ip-address/hostname>:8080/erwinDISuite/saml2/service-provider-metadata/{registerID}

   Update the above URL with the same register ID configured in Step1.

   Example:

   https:// 10.222.16.5:8080/erwinDISuite/saml2/service-provider-metadata/MSAD

   It automatically downloads the Service Provider Metadata.xml

   (File name: saml-MSAD-metadata.xml)

2. If SAML has already been configured on erwin DI previously, then:

   Open the browser and navigate to below URL to download SAML Service Provider Metadata.xml

   https://<ip-address/hostname>:8080/erwinDISuite/saml/metadata

   Example:

   https:// 10.222.16.5:8080/erwinDISuite/saml/metadata

   It automatically downloads the Service Provider Metadata.xml

   (File name: saml-MSAD-metadata.xml)

3. The file looks like below.

4. Once spring_saml_metadata.xml is generated, **provide this file to the customer's SAML administrator for Idp.xml file generation**.

Step 3: Customer's SAML administrator to generate idp.xml

1. Once spring_saml_metadata.xml is provided to customer's SAML administrator, he/she needs to generate Idp.xml based on SAML metadata.
2. Let your SAML administrator know that below are the attributes that need to be added in the SAML Response [Please note that the below attributes are case sensitive]
    a. FirstName
    b. LastName
    c. EmailAddress
    d. amm_user_role
3. Once idp.xml is generated, customer needs to send it back to client side erwin administrator

Step 4: Drop idp.xml in erwin DI Suite tomcat root folder

- Once idp.xml is received, copy the file and paste it in the metadata folder in the following path

(C:\Program Files\Apache Software Foundation\Tomcat\webapps\erwinDISuite\WEB-INF\classes\resources\metadata)

Step 5: Restart the erwin DI application from the tomcat console

Test for successful Single Sign-on (SSO) configuration once all the above steps have been implemented.

Frequently asked questions

- **Can customers use their own security certificate .crt and private key .key files instead of erwin default local.crt and local.key files?**
  Yes, customers can use their own security certificate .crt and private key .key files if they want to. However, the files should be placed in the correct path and the same file name needs to be configured in database.properties file.

  (as explained in Step1 in the guide)

```
############################################################
spring.security.saml2.customrelyingparty.samlentityid=ErwinDev
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].certificate-location=classpath:resources/Quest.crt
spring.security.saml2.relyingparty.registration.MSAD.signing.credentials[0].private-key-location=classpath:resources/QSoft.key
spring.security.saml2.customrelyingparty.algorithm=http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
spring.security.saml2.customrelyingparty.samlrolesseparator=$
#saml configuration ends
```

- **Can erwin team generate idp.xml?**
  No. This needs to be done by the customer's SAML administrator only.

- **Who are the supported Idp providers?**
  Refer to section 4 for the list of Supported SAML authentication providers. However, erwin Data Intelligence Suite is compatible with all external IdPs that support SAML 2.0.

- **What are the parameters/attributes used by erwin Data Intelligence Suite during authentication?**
  - Refer to Step3 - These are the type of attributes that need to be added in the SAML Response - FirstName, LastName, EmailAddress, amm_user_role.
  - SAML administrator needs to share/add these details along with all requests.
  - Please note that these parameters/attributes are case sensitive.

- **Unable to login to erwin DI using SAML even after enabling/configuring the required roles?**
  Please make sure that all the roles used in the SAML parameter amm_user_role are present in the erwin DI application. In case they are not present then the SAML login will not work.

- **How do I get the logs for any SSO related issue?**
  - We can investigate the tomcat application logs (found within the tomcat\logs folder where the application is installed)
  - Additionally, you can download the *Chrome plug-in - SAML Tracer* to capture more logs. Download it from the Chrome store as a browser plugin. Please create an erwin DI support ticket with all exported logs to further debug the issues.