

## Single Sign-On (SSO) configuration Guide

# Quest Data Intelligence (DI)

**Version 16.0**

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

© 2026 Quest Software Inc. ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Quest Software Inc.

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

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**

Quest, the Quest logo, Quest Data Intelligence, erwin by Quest are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit <https://www.quest.com/legal/trademark-information.aspx>. All other trademarks and registered trademarks are property of their respective owners.

## Contents

---

<b>About this Guide .....</b>	<b>4</b>
<b>Introduction .....</b>	<b>4</b>
<b>Pre-Requisites .....</b>	<b>4</b>
<b>Supported SAML authentication providers.....</b>	<b>4</b>
<b>Configuration steps .....</b>	<b>5</b>
Step 1: Configure SAML Properties in database.properties file .....	5
Step 2: Generate Service Provider Metadata.xml.....	8
Step 3: Customer’s SAML administrator to generate idp.xml.....	9
Step 4: Drop idp.xml in Quest DI Suite tomcat root folder .....	9
Step 5: Restart the Quest DI application from the tomcat console.....	9
<b>Frequently asked questions .....</b>	<b>10</b>

## 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 Quest Data Intelligence.

## Introduction

---

The identity federation standard Security Assertion Markup Language (SAML) 2.0 enables the secure exchange of user authentication data between Quest Data Intelligence 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). Quest Data Intelligence, acts as the service provider (SP).

## Pre-Requisites

---

- This configuration guide is developed considering that SAML is already configured within customer's organization, and Quest Data Intelligence 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 Quest 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

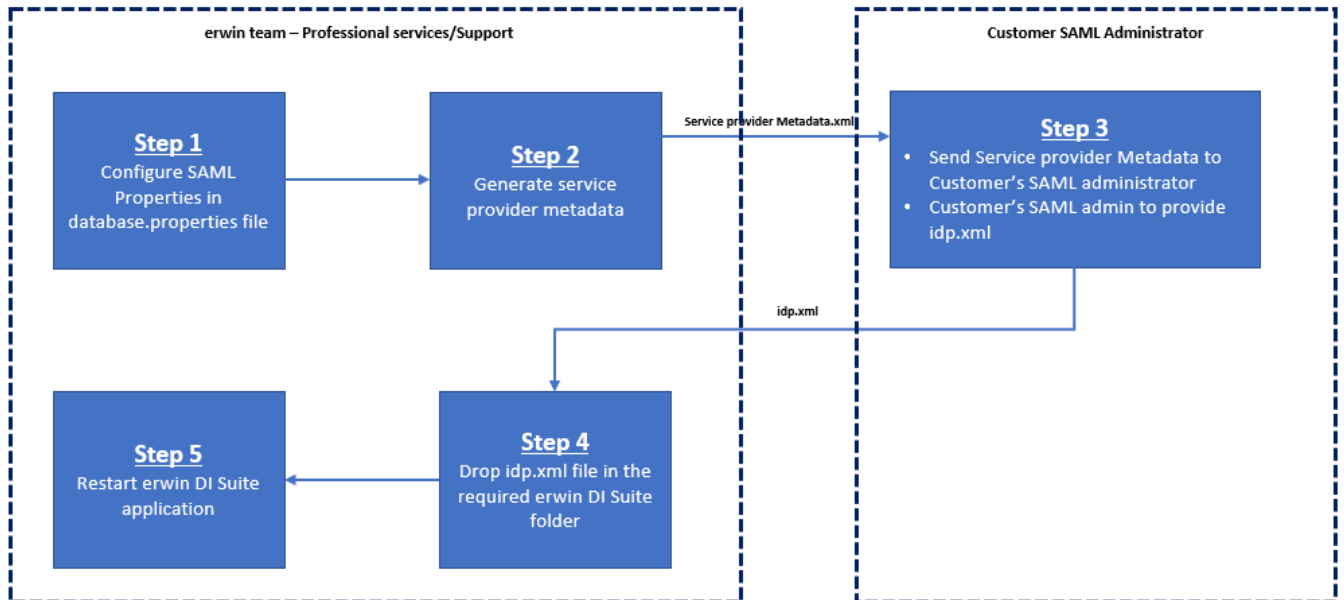
---

Quest 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, Quest 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\QuestDI\WEB-INF\database\database.properties
2. Update WebAuthenticationMechanism from “DB” to “DB, SAML”

```

### Authentication Configuration Begin
WebAuthenticationMechanism=DB, SAML
WebServiceAuthenticationMechanism=DB
### Authentication Configuration End
    
```

3. Steps to update SAML configuration section parameters

### Step 3.1

spring.security.saml2.customrelyingparty.samlentityid

- If you are configuring SAML for Quest 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 Quest DI previously then update this parameter to the same value of SAMLEntityId parameter present in your old database.properties file [Prior to Quest DI 13.2]

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

Eg: `spring.security.saml2.customrelyingparty.samlentityid= QuestDev`

### 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\QuestDI\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\QuestDI\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 Quest 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.samlroleseparator = ^`

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

Example with role separator as Dollar symbol \$

`spring.security.saml2.customrelyingparty.samlroleseparator =$`

### Sample SAML configuration for first time SAML Quest 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/2001/04/xmldsig-more#rsa-sha256  
spring.security.saml2.customrelyingparty.samlroleseparator=$  
#saml configuration ends
```

### Sample SAML configuration for clients who have SAML configured on Quest 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/2001/04/xmldsig-more#rsa-sha256  
spring.security.saml2.customrelyingparty.samlroleseparator=$  
#saml configuration ends
```

### Step 3.6

Restart the QuestDI web application from tomcat console.



## 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 Quest DI 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\QuestDI\WEB-INF\classes\resources\metadata)

## Step 5: Restart the Quest 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 Quest 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 Quest 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, Quest Data Intelligence is compatible with all external IdPs that support SAML 2.0.
- **What are the parameters/attributes used by Quest Data Intelligence during authentication?**
  - Refer to Step3 - These are the 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 Quest 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 Quest 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.

**Note:** If you encounter any issues, reach out to our Support team with all the log files available so we can troubleshoot further.