# Table of Contents

Security Assertion Markup Language (SAML) is a standard for logging users into applications based on their sessions in another context. This single sign-on (SSO) login standard has significant advantages over logging in using a username and password. SAML is very powerful and flexible, but the specification can be quite a handful.

## Steps to enable SAML SSO Authentication

- See this article for details

## Tracking logins

You can see how your remote authentication works in the KBPublisher login logs, located under **Logs > Logins**.

For debugging, the most recent remote login is logged to a file called */last_remote_login.log* in the KBPublisher cache directory (*APP_CACHE_DIR* in *admin/config.inc.php*). For example: */home/username/kb_cache/last_remote_login.log*

The following steps assume that you have an account with a supported identity provider. You need to know the SAML Login URL and have the x.509 certificate supplied by your identity provider. You should also be familiar with the format of the SAML identity response from your SAML provider.

## Enabling SAML authentication

1. As a user with administrator privileges, go to **Settings → Authentication**.

2. On the **SAML** tab, check **Enable Single Sign-On**
   (make sure that $conf['auth_remote'] in the file *admin/config.inc.php* is set to 1)

3. Select an **Authentication Mode**, which defaults to *Only SAML Authentication allowed*. The available options are:
   - **Both built-in and SAML Authentication allowed** - User will be able to log in to KBPublisher using SSO and/or built-in authentication.
   - **Only SAML Authentication allowed** - User can only log in using SSO.
   - **Only SAML Authentication allowed, try auto authentication** - User can only log in using SSO and, if possible, auto-authentication on the SAML server will be applied.

4. In the **Multi-Factor Authentication** field, choose *The same as MFA Policy* setting if you want the built-in MFA to be available for SAML authentication. You can select the desired MFA policy in **Admin → Security/Privacy → Multi-Factor Authentication**.

5. Enter the following SAML Configuration settings:
   - **Name** - The name for your SSO provider, to be presented on the login page.
   - **Issuer** - Unique identifier for your Identity Provider (typically a URL). In some cases, this is called the Entity ID.
   - **Single Sign-On Service Url** - The SAML Login URL where the Controller will route Service Provider (SP)-initiated login requests. This is required.
   - **Single Logout Service Url** - The URL where the Controller will redirect users after they log out. If you do not specify a logout URL, users will get the KBPublisher login screen when they log out.
   - **Single Sign-On Service Binding** and **Single Logout Service Binding** - You can change the binding for login and logout URLs. Defaults to HTTP-Redirect. The other option is HTTP-POST.
   - **Public X.509 Certificate** - The x.509 certificate from your identity provider configuration.
   - **Authentication Contexts -** Set possible auth context values, place a value on a new line. Leave it empty for default value.
     Example:
     urn:oasis:names:tc:SAML:2.0:ac:classes:Password
     urn:oasis:names:tc:SAML:2.0:ac:classes:X509
     If empty It defaults to: *urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport*
     For Azure AD set to: *urn:oasis:names:tc:SAML:2.0:ac:classes:Password*

6. In the SAML **User Mapping Fields** settings, specify how SAML-authenticated users are identified in the KBPublisher Controller:
   - **Remote User Id** - Unique identifier for the user in the SAML response. This value is responsible to identify user in authentication requests. It defaults to the SAML NameID element.
   - **First Name** - The first name for the user corresponding to the KBPublisher First Name field. Given the sample response, this value would be User.firstName.
   - **Last name** - The last name for the user corresponding to the KBPublisher Last Name field. Given the sample response, this value would be User.lastName.
   - **Email** - The user's email address, corresponding to KBPublisher email field. The value must be unique among all SAML users. Given the sample response, this value would be User.email.
   - **Username** - This value corresponds to the KBPublisher username field. The value must be unique among all SAML users. Given the sample response below, the value for this setting would be User.email.

   SAMPLE RESPONSE

```
<saml:AttributeStatement>
  ...
  <saml:Attribute Name="User.OpenIDName" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">adynamo</saml:AttributeValue>
  </saml:Attribute>
  ...
  <saml:Attribute Name="User.firstName" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">John</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="User.lastName" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">Doo</saml:AttributeValue>
  </saml:Attribute>
  <saml:Attribute Name="User.email" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xsi:type="xs:string">John.Doo@example.com</saml:AttributeValue>
  </saml:Attribute>
  ...
</saml:AttributeStatement>
```

7. To map SAML group attributes to KBPublisher privileges and roles, configure the **SAML attribute for Privileges** and **SAML attribute for Roles** settings. The settings you use depends on the structure of the SAML group attribute in the response.

   **Note:** You can skip this step. If you leave these settings empty they will never be rewritten on user login. You can assign required privileges and/or roles later in KBPublisher.

   1. Click the ellipsis button in the **SAML attribute for Privileges** field to open the Group-to-Privilege Mapping dialog box.
   2. Enter the *SAML group attribute name*, the *SAML group attribute value*, and choose a *KBPublisher privilege* to map to. In the sample response provided below, we map group User.group with value Editor.
   3. Click **Add**.
   4. When you have added all mapping rules, click **Done**.

   SAMPLE RESPONSE

```
<saml:AttributeStatement>
  ...
  <saml:Attribute Name="User.OpenIDName" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">adynamo</saml:AttributeValue>
  </saml:Attribute>
  ...
  <saml:Attribute Name="User.group" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:basic">
    <saml:AttributeValue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xs:string">Editor</saml:AttributeValue>
  </saml:Attribute>
  ...
</saml:AttributeStatement>
```

   **Attention:** If you map SAML groups to KBPublisher privileges, all matched users will be assigned the specified privilege. If you do not have an Unlimited license and the number of allowed staff users is exceeded, the privilege will not be assigned to the user.

   Use the same steps to map **SAML attribute for Roles**.

8. Set up **Service Provider** values.
   - **Metadata** - Use this information to register the Knowledgebase with your identity provider.
   - **Public X.509 Certificate** - If your identity provider requires signing and/or encryption, copy the contents of your certificate and paste it here.
   - **Private Key** - If your identity provider requires signing and/or encryption, copy the contents of your private key and paste it here.
   - **Signing algorithm** - Select a signing method for all SAML requests.

## Additional configurations

- **Rewrite user on login** - This is the time, in seconds, to rewrite user data on login. Enter *0* to disable updates to user data on login. Enter *1* to rewrite user data on every authentication request.
- **User account info** - This field indicates whether or not the user is able to update his account info. Available options include:
  - **0** - OFF, user can't update his account info
  - **1** - ON, user can update his account info
  - **2** - AUTO, depends on other remote settings

1. Log into your OneLogin account.

2. Go to the "Apps" section and then click the "Add app" button.
   In the filter box type "SAML" and select the "SAML Test Connector (IdP w/attr)" application.
   Change the display name of your app if you wish and click "Save".

3. In your KBPublisher administrator area go to Settings -> Authentication Provider -> SAML.
   Find the "Service Provider" section and open the "Metadata" window.
   On the Configuration tab of your Onelogin app, match the following settings:
   Audience: [Entity Id]
   Recipient: [Entity Id]
   ACS (Consumer) URL Validator: .*.
   ACS (Consumer) URL: [Assertion Consumer Service URL]
   Single Logout URL: [Single Logout Service URL]

   Click on the "Save" button.

4. KBPublisher settings for attributes names default to Onelogin attributes, but you might need to ensure they are equal.

5. Go to the "SSO" tab, then click on the "View Details" link at the "X.509 Certificate" field.
   Copy the certificate contents and paste it into the "Public X.509 Certificate" window in KBPublisher.
   Copy the values of "Issuer URL", "SAML 2.0 Endpoint (HTTP)" and "SLO Endpoint (HTTP)" and paste them into the corresponding KBPublisher fields as shown below.

6. If you don't already have a user on OneLogin go to the "Users" tab and add one.
   Go to the "Applications" tab on the user page and assign the application to him.

7. On your KBPublisher settings page click the "Test / Debug" button.
   If it works, you'll see a greeting message.

8. If click on "App Icon" in Onelogin (or Okta) to login to KBPublisher does not work, fill RelayState field in app settings, it should be full path to your KBPublisher installation (example: https;//domain.com/kb/)