

# Steps to Configure the ASP.NET SAML 2.0 Module

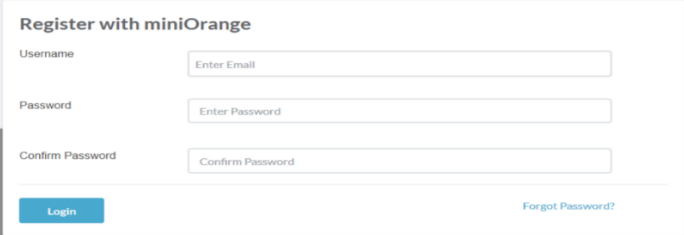
---

## Step 1: Download and extract package.

- Download miniOrange ASP.NET SAML 2.0 Module.
- For Setting up the module, extract the **asp-net-saml-sso-module-xxx.zip**, you will find a DLL file **miniorange-saml-sso.dll**, a configuration file **saml.config** and a **integration.md** file which contain the steps for adding the module into your application.

## Step 2: Add the module in your application.

- Add **miniorange-saml-sso.dll** in the bin folder (where your other DLL files exist) for your application.
- Register **miniorange saml sso** module in your application according to the provided steps in the **integration.md** file.
- Add the provided configuration file **saml.config** in the root directory for your application.
- After integration, open the browser and browse the connector dashboard with the URL below: [https:// <your-application-base-url>/?ssoaction=config](https://<your-application-base-url>/?ssoaction=config)
- If it pops up the registration page or login page, you have successfully added the miniOrange saml sso connector for your application.
- **Register or Login** for configuring the module.



**Register with miniOrange**

Username

Password

Confirm Password

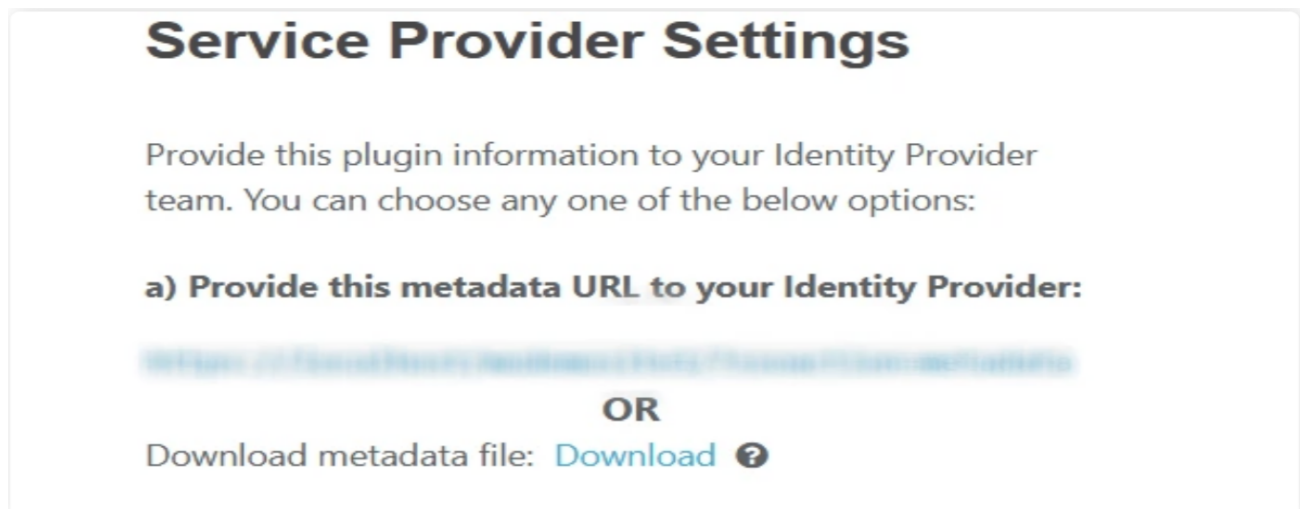
[Login](#) [Forgot Password?](#)

### Step 3: Configure your Identity Provider

There are below two ways with which you can get the SAML SP metadata to configure on your identity provider end.

#### A] Using SAML metadata url or metadata file:

- Under the **Service Provider Settings** section you can find the metadata URL as well as option to download the SAML metadata.
- Copy metadata URL or download metadata file to configure the same on your identity provider end.
- You can refer to the below screenshot:



#### B] Uploading metadata manually:

- From the **Service Provider Settings** section you can copy the service provider metadata manually like **SP Entity ID**, **ACS Url**, **Single Logout Url** and provide it to your identity provider for configuration.
- You can refer to the below screenshot:

## b) Provide the following information to your Identity Provider

Base Url

SP Entity ID

ACS Url

Single Logout Url

### Step 4: Configure your Service Provider

There are below two ways with which you can configure your SAML identity provmetadata in the module.

#### A] Upload metadata using the Upload IDP Metadata button:

- If your identity provider has provided you with the metadata URL or metadata file (.xml format only) then you simply configure the identity provider metadata in the module using the **Upload IdP Metadata** option.
- You can refer to the below screenshot:

The screenshot shows a dialog box titled "Upload IDP Metadata" with a close button (X) in the top right corner. The dialog is divided into two sections by the word "OR".

The first section is for uploading a file. It contains the text "Upload XML File:" followed by a "Browse..." button and the text "No file selected.". To the right of this section is a blue "Upload" button.

The second section is for entering a URL. It contains the text "Enter metadata URL:" followed by a text input field. Below the input field is a blue "Fetch Metadata" button.

- You can choose any one of the options according to the metadata format you are available with.

## B] Configure the identity provider metadata manually:

- After configuring your **Identity Provider**, it will provide you with **IDP Entity ID**, **IDP Single Sign On URL** and **x.509 Certificate**. Configure these values under IDP Entity ID, Single Sign-On Url and SAML X509 Certificate fields respectively.
- Click **Save** to Save your IDP details.

ASP.NET SAML Connector

⚙️ Plugin Settings

🔍 Login Setup

📄 Integration Code

📄 Premium Plans

👤 Account Setup

🛠️ Support

### Identity Provider Settings

[Upload IDP Metadata](#)

\* IDP Entity ID

\* Single Sign-On Url

SAML Login Binding types  
 Http-Redirect  Http-Post

SAML Logout Binding types  
 Http-Redirect  Http-Post

SAML Logout Url

SAML X509 Certificate

Auto-Redirect to IDP  
 Signed Login Request  
 Signed Logout Request  
 Header Based Authentication  
 Cookie Based Authentication

[Save](#) [TestConfiguration](#)

### Service Provider Settings

Provide this plugin information to your Identity Provider team. You can choose any one of the below options:

**a) Provide this metadata URL to your Identity Provider:**

<https://example.com/metadata/ExampleServiceProviderMetadata>

OR

Download metadata file: [Download](#)

**b) Provide the following information to your Identity Provider**

Base Url

SP Entity ID

ACS Url

Single Logout Url

[Save](#)

### Attribute Mapping

Username

Email

Firstname

Lastname

### Custom Attribute Mapping

Attribute Name	Attribute Value	+
----------------	-----------------	---

[Save Attribute Mapping](#)

## Step 5: Test Configuration

- Click on the **Test Configuration** button to test whether the SAML configuration you've done is correct or not.
- The below Screenshot shows a successful result:

**Test Configuration**

Attribute Name	Attribute Value
NameID	avnash@cs.cmu.edu
firstname	Avnash
lastname	Maik
email	avnash@cs.cmu.edu

Done

## Step 6:Attribute Mapping

- Map your application attributes with the Identity Provider (IdP) attributes.
- **Note: All the mapped attributes will be stored in the session so that you can access them in your application.**

### Attribute Mapping

NameID

Email

Firstname

Lastname

### Custom Attribute Mapping

<input type="text" value="Attribute Name"/>	<input type="text" value="Attribute Value"/>	<input type="button" value="+"/>
---	--	----------------------------------

## Step 7: Integration Code

- You can find the **integration code** in the Integration Code tab in the connector. Just copy-paste that code snippet wherever you want to access the user attributes.

### Prefix For Session Attributes

(Session Attribute's name will be prefixed with below string.)

You can access user attributes in session as below. Just copy & paste below code snippet to your application for accessing the attributes from the session.

```
string username; string email; string firstname; string lastname;
if(Session["saml_sso_username"]!=null)
{
    username=Session["saml_sso_username"].ToString();
}
if(Session["saml_sso_email"]!=null)
{
    email=Session["saml_sso_email"].ToString();
}
```

Save

### Step 8: Add following link in your application for Single Sign-On (SSO)

- Use the following URL as a link in your application from where you want to perform SSO:

<https://base-url/?ssoaction=login>

- For example you can use it as:

```
<a href="https://base-url/?ssoaction=login">Log in</a>
```

### Step 9: Add following link in your application for SLO

- Use the following URL as a link in your application from where you want to perform SLO: <https://base-url/?ssoaction=logout>

- For example you can use it as:

```
<a href="https://base-url/?ssoaction=logout">Log out</a>
```

You can configure the ASP.NET SAML 2.0 Single Sign-On (SSO) module with any Identity Provider such as **ADFS, Azure AD, Bitium, Centrify, G Suite, JBoss Keycloak, Okta, OneLogin, Salesforce, AWS Cognito, OpenAM, Oracle, PingFederate, PingOne, RSA SecureID, Shibboleth-2, Shibboleth-3, SimpleSAML, WSO2** or even with your own

**custom identity provider** and secure your ASP.NET / ASP.NET MVC / ASP.NET Core / VB.NET applications by performing Single Sign-on (SSO) into them.

Not able to find your IdP? Contact us at [info@xecurify.com](mailto:info@xecurify.com) and we'll help you set up SSO with your IdP in no time.

The miniOrange ASP.NET SAML 2.0 Connector supports **C# and Visual Basic** languages.

**We also provide DotNet Nuke Single Sign-On SAML Connector. [Click Here to know more.](#)**

**miniOrange also provides you modules for integration with legacy apps such as Active Directory, SiteMinder, Radius, Unix and so on.**

**miniOrange helps you for adding login/authentication into your ASP.NET site using OAuth, OpenID, ADFS, SAML and WSFED protocols.**

**Have questions or need clarifications? Please contact us at [info@xecurify.com](mailto:info@xecurify.com) and our experts will be there to help you.**

**Reference links:**

- [Introduction to ASP.NET SAML Single Sign-On \(SSO\)](#)
- [Single Logout- How it works](#)
- [Single Sign-On \(SSO\) to an ASP.NET mvc application](#)
- [Single Sign-On \(SSO\) to an ASP.NET core application](#)