Steps to Configure the ASP.NET SAML 2.0 Module
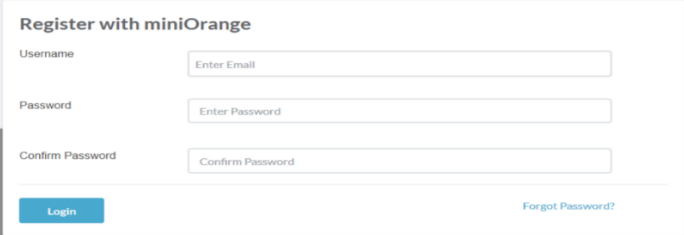
Step 1: Download and extract package.

- 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`
- 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.
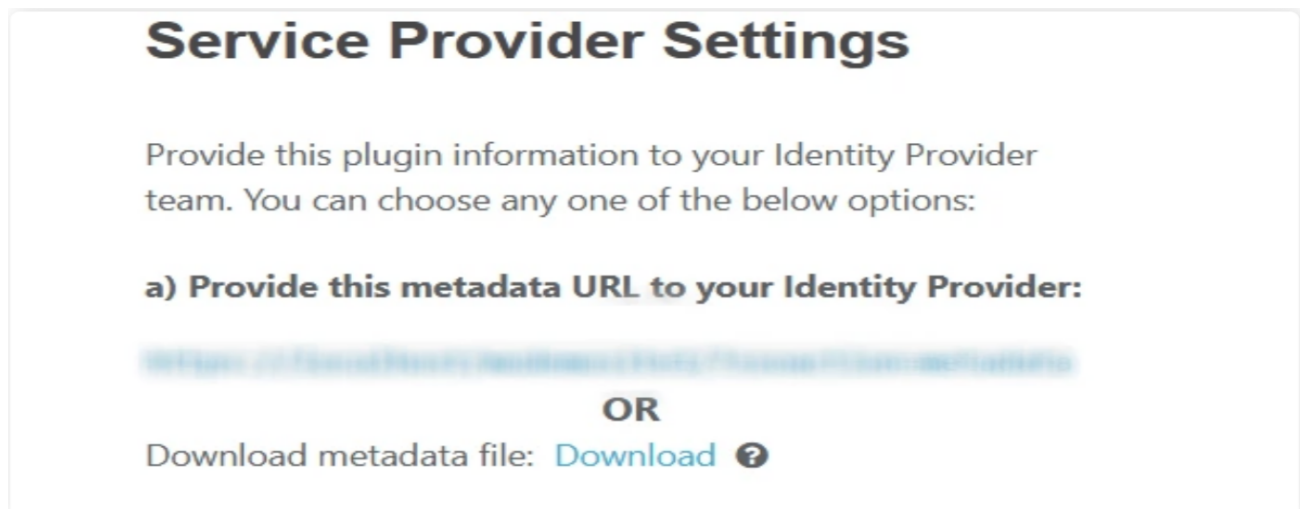
Step 3: Configure your Identity Provider

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:

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:

OR

Download metadata file: Download ❓

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:

Upload IDP Metadata ✕

Upload XML File: [Browse...] No file selected. [Upload]

OR

Enter metadata URL:

[Fetch Metadata]

- 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.

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	a̶m̶a̶s̶h̶@̶m̶u̶a̶g̶e̶.̶c̶o̶m̶
firstname	Avinash
lastname	Naik
email	m̶a̶s̶h̶@̶r̶i̶c̶n̶g̶.̶c̶o̶m̶

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

NameID

Email

Email

Firstname

Firstname

Lastname

Lastname

Custom Attribute Mapping

Attribute Name | Attribute Value | +

Save Attribute Mapping

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.

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 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 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**