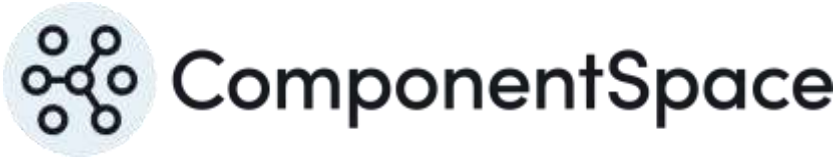

ComponentSpace
SAML for ASP.NET Core
Step by Step Guide
Identity Provider

Contents

Introduction

This document walks you through the recommended steps for enabling your web application to act as an identity provider and support SSO with partner service providers.

More detailed information about the SAML API, SAML configuration and related topics is available in the various documents listed in the Quick Start Guide.

Creating a Certificate

An X.509 certificate and associated private key are required as SAML messages or assertions sent by your identity provider should be signed.

Use the CreateSelfSignedCert console application to create a self-signed certificate. For more information, refer to the Certificate Guide.

```
dotnet CreateSelfSignedCert.dll
```

Neither the private key file nor its password should be shared with third parties.

Copy these certificate files to a certificates folder under your application.

Creating the Local Identity Provider Configuration

SAML configuration is used to specify the local identity provider. The Configuration Guide describes the various alternatives for specifying SAML configuration. Here we will use the simplest approach which is to store the SAML configuration in your application's appsettings.json.

Use the CreateConfiguration console application to create a saml.json which may be merged into your application's appsettings.json. For more information, refer to the Configuration Guide.

```
dotnet CreateConfiguration.dll
```

Add the generated configuration to your application's appsettings.json.

Exporting the Local Identity Provider Metadata

SAML metadata is the standard format for exchanging configuration information between SAML providers. SAML metadata is supplied to partner providers so they can update their internal configuration to support SSO.

Use the ExportMetadata console application to generate the SAML metadata. For more information, refer to the SAML Metadata Guide.

```
dotnet ExportMetadata.dll
```

Share the SAML metadata with your partner provider(s). You could make the metadata available for download from a URL or supply it in an email etc.

Importing the Partner Service Provider Metadata

SAML metadata supplied by partner providers is used to update your SAML configuration.

Use the ImportMetadata console application to update the SAML configuration. For more information, refer to the SAML Metadata Guide.

```
dotnet ImportMetadata.dll
```

Updating the Application Code

The Developer Guide describes the various SAML APIs to support SSO and SLO flows when acting as the identity provider.

The Examples Guide walks through the SAML specific code in the example identity provider.

Use these guides to enable SAML SSO in your application.

Testing SAML SSO

Before testing, ensure the following have been completed:

1. Your application's appsettings.json has been updated with the SAML configuration.
2. The local and partner certificate files are located in your application's certificates folder.
3. Your application has been updated to call the SAML API.
4. The partner provider has imported your SAML metadata and is ready.