ComponentSpace OpenID Connect for ASP.NET Core Installation Guide

Contents

Introduction

The ComponentSpace OpenID Connect for ASP.NET Core is a .NET class library that provides OpenID provider functionality.

You can use this functionality to easily enable your ASP.NET Core web applications to participate in OpenID Connect flows as the OpenID provider. Example applications with full source code are included.

Installation

The product is supplied as a ZIP file. This includes the binaries, documentation, and example projects. Simply unzip to a convenient location on your file system (eg C:\OpenID for ASP.NET Core).

The product does not need to be installed on production or test servers. Instead, the OpenID Connect for ASP.NET Core class library should be deployed along with the application.

NuGet Package Source

The ComponentSpace.OpenID DLL is shipped as a NuGet package. This is found in the bin folder and its name format is:

ComponentSpace.OpenID.<version-number>.nupkg

Note that an evaluation version of the product is published through nuget.org.

During evaluation, the nuget.org source may be used. Alternatively, a local package source may be specified in Visual Studio.

For general information on NuGet packages and their consumption, refer to:

https://docs.microsoft.com/en-us/nuget/consume-packages/overview-and-workflow

For information on hosting packages, refer to:

https://docs.microsoft.com/en-us/nuget/hosting-packages/overview

For information on using Azure Artifacts in an Azure DevOps environment, refer to:

https://docs.microsoft.com/en-us/azure/devops/artifacts/get-started-nuget

NuGet.config

A NuGet.config file is included with the examples' solution. It specifies the bin folder as a package source.

For new solutions, a similar NuGet.config file may be used to specify the package source for that solution.

NuGet includes several options for specifying the package source including project-specific, solution-specific, and machine-wide NuGet.config files. The appropriate file should be updated to specify the package source.

For more information, refer to:

https://docs.microsoft.com/en-us/nuget/consume-packages/configuring-nuget-behavior

NuGet Package Manager

Rather than directly updating NuGet.config, the package source may be specified through Visual Studio.

To specify a package source:

1. From Visual Studio's menu, select Tools > Options.
2. Under the listed options, select NuGet Package Manager > Package Sources.
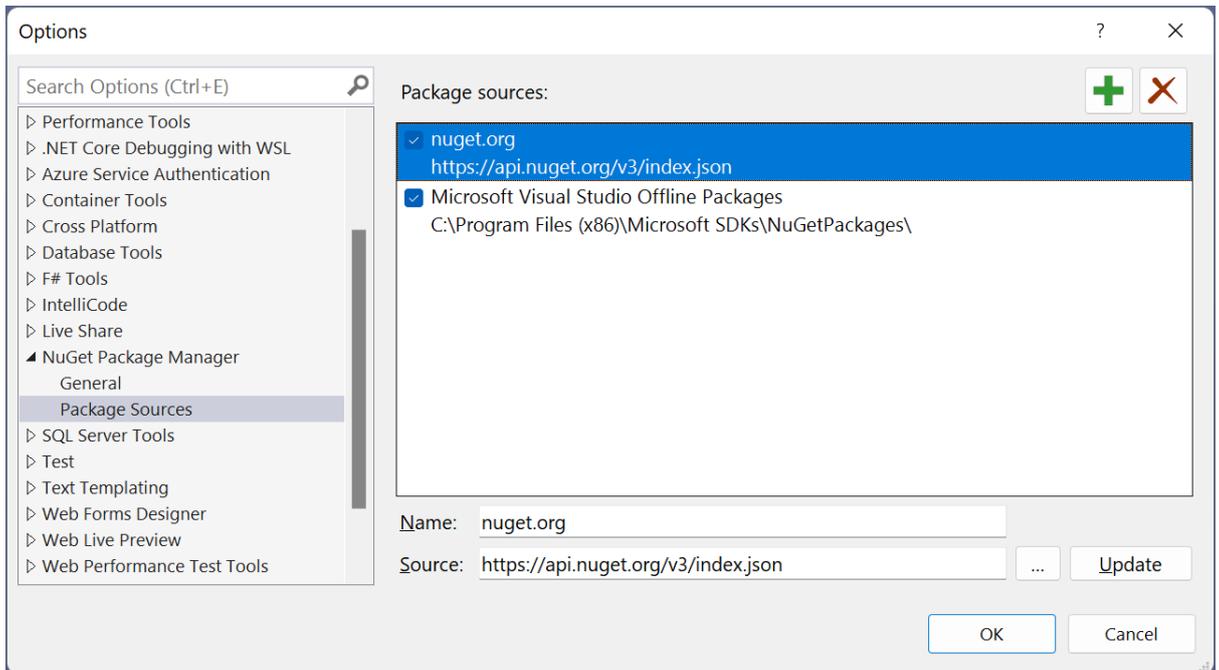
Figure 1 Package Sources

3. Add a package source specifying the bin folder where the ComponentSpace.OpenID nupkg is located. This package source is available to all ASP.NET Core web application projects.
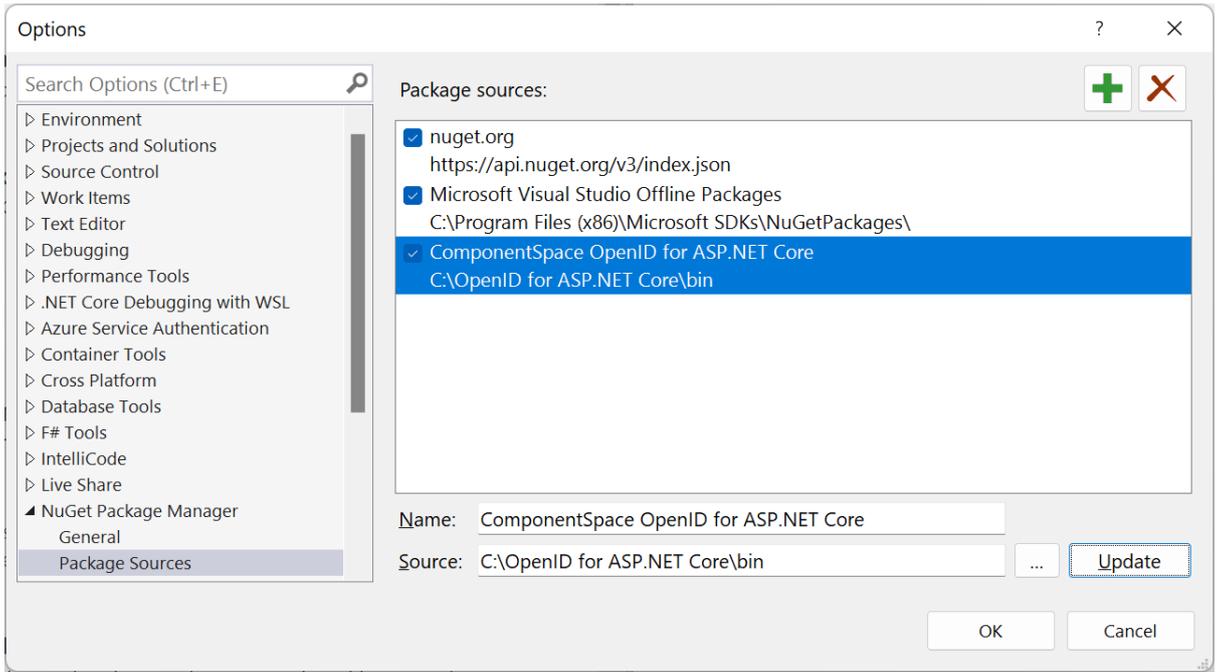
Figure 2 Add ComponentSpace Package Source

NuGet Package Installation

1. Within an ASP.NET Core web application project, click Manage NuGet Packages and select ComponentSpace as the package source.

Figure 3 Select ComponentSpace Package Source

2. Browse and install ComponentSpace.OpenID into the application.
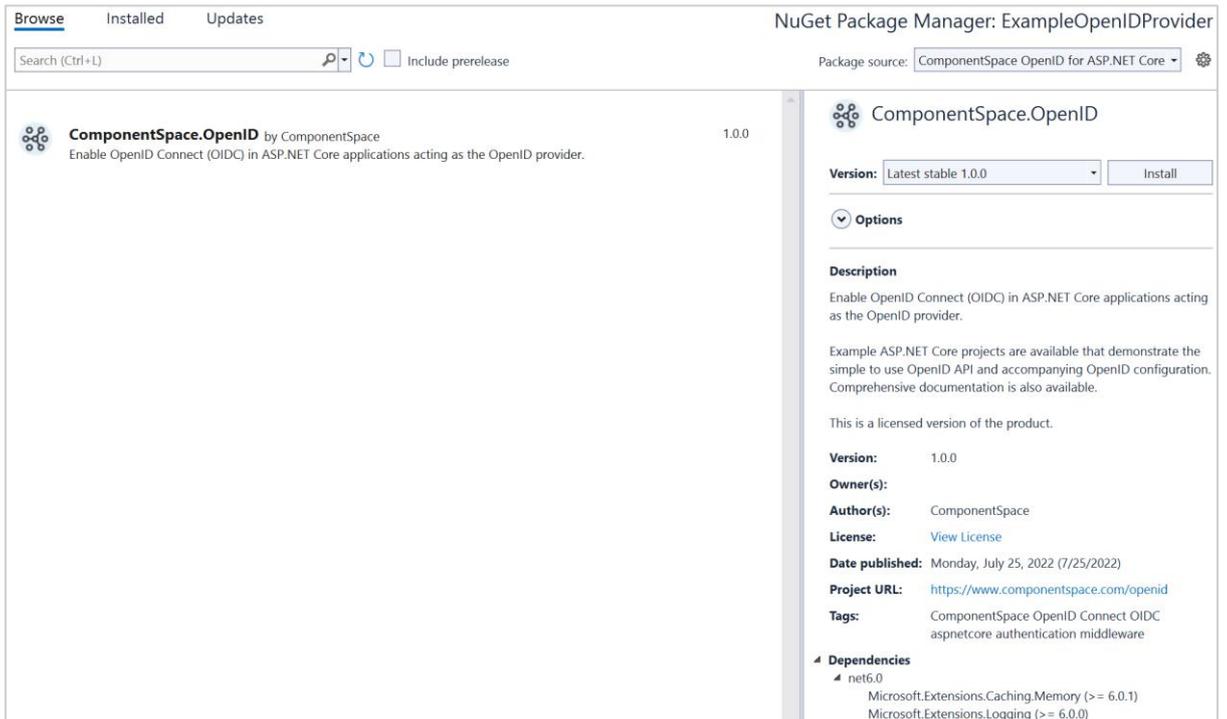
Figure 4 Install ComponentSpace.OpenID

3. Accept the license agreement.
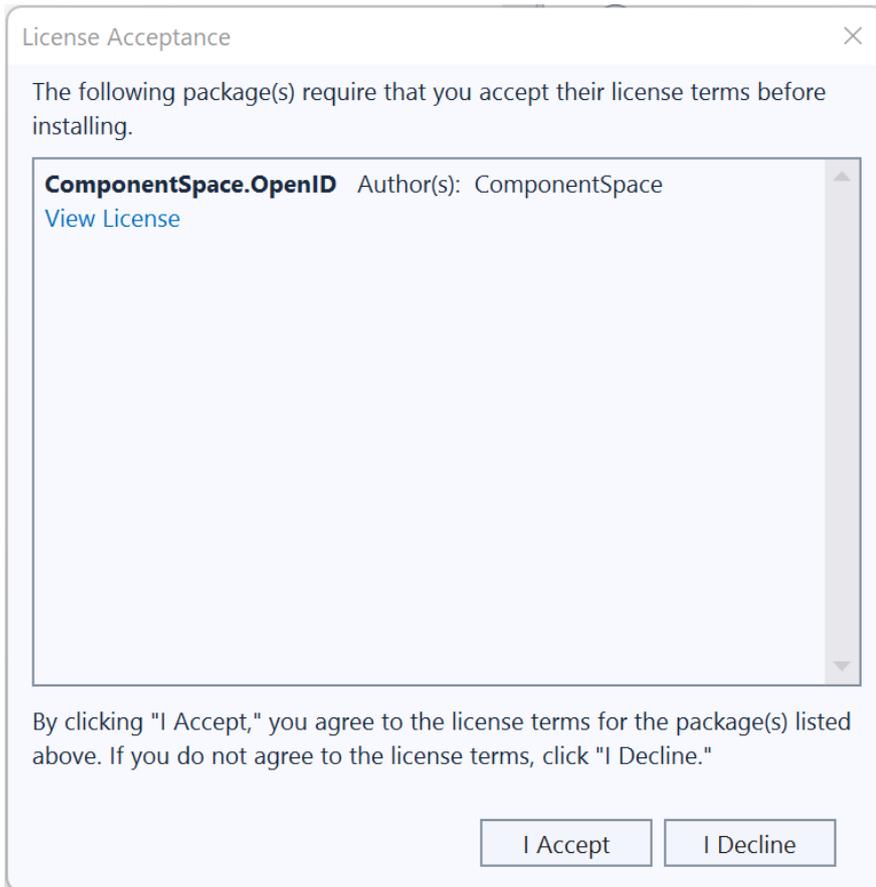
Figure 5 License Agreement Acceptance

4. Visual Studio will install the package and its dependencies. This can be confirmed by ensuring Intellisense recognizes the namespace ComponentSpace.OpenID.

Evaluation vs Licensed Version

An evaluation version of the product is equivalent to the licensed version except that it will stop working once the evaluation period expires.

The package name of the evaluation version is **ComponentSpace.OpenID**.

The package name of the licensed version is **ComponentSpace.OpenID.Licensed**.

Different package names are used to avoid accidentally upgrading from a licensed to an evaluation version of the product.

An evaluation version of the ComponentSpace.OpenID DLL includes the word evaluation. The licensed version doesn't.
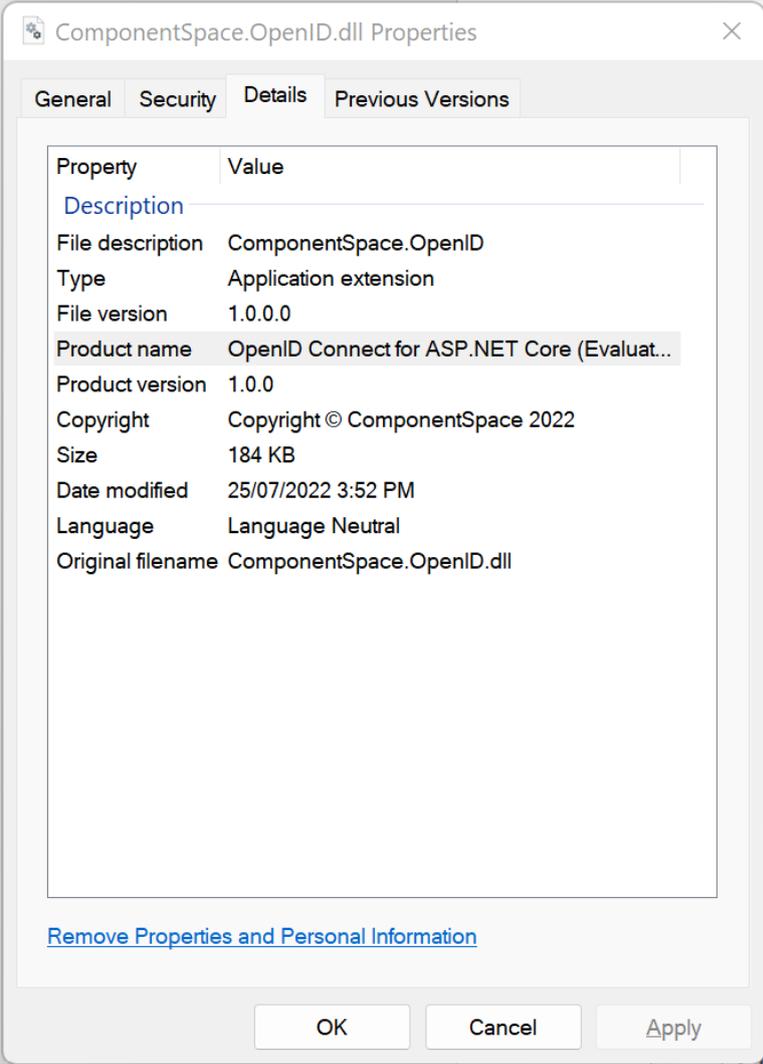
Figure 6 Evaluation Version DLL Properties

Also, the package description specifies whether it's an evaluation or licensed version.
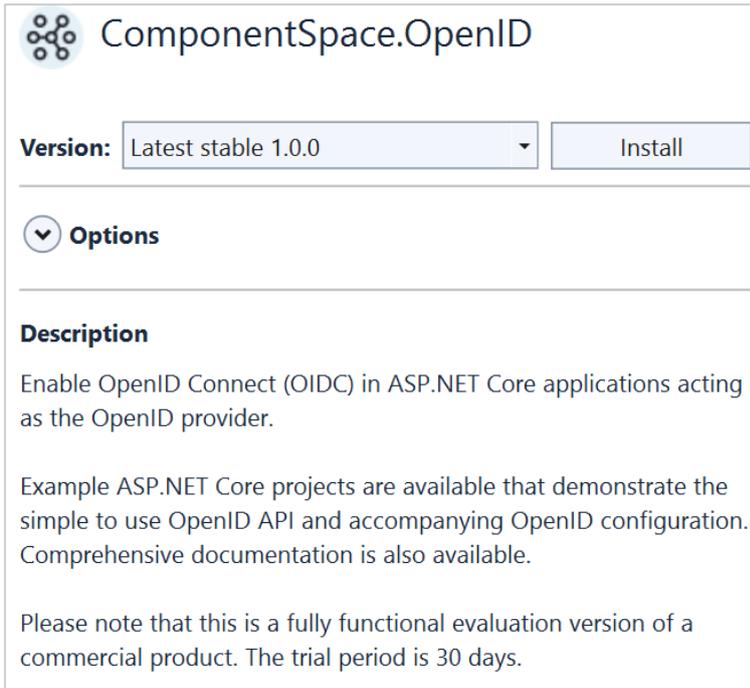
Figure 7 Evaluation Version Package Description

Upgrading to a Licensed Version

After purchasing a licensed version of the product, ensure all time-limited evaluation packages (i.e. ComponentSpace.OpenID) are replaced with the licensed package (i.e. ComponentSpace.OpenID.Licensed).

Uninstall the **ComponentSpace.OpenID** package from your applications.
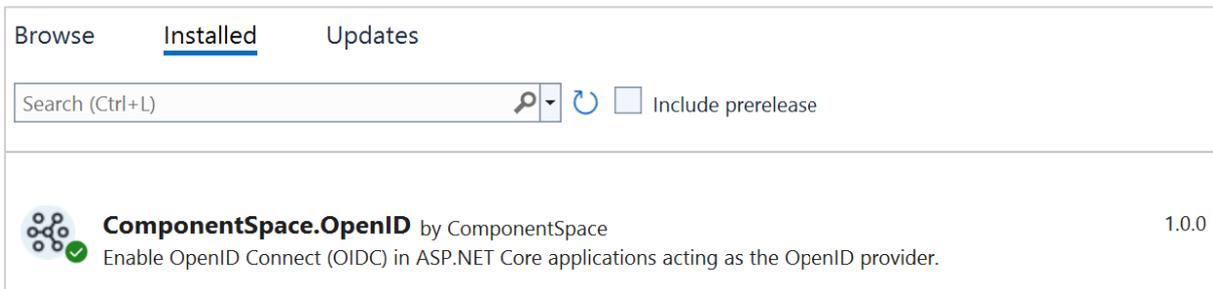
Figure 8 Evaluation Version Installed

Setup a NuGet package source for the licensed package.

Install the **ComponentSpace.OpenID.Licensed** package into your applications.
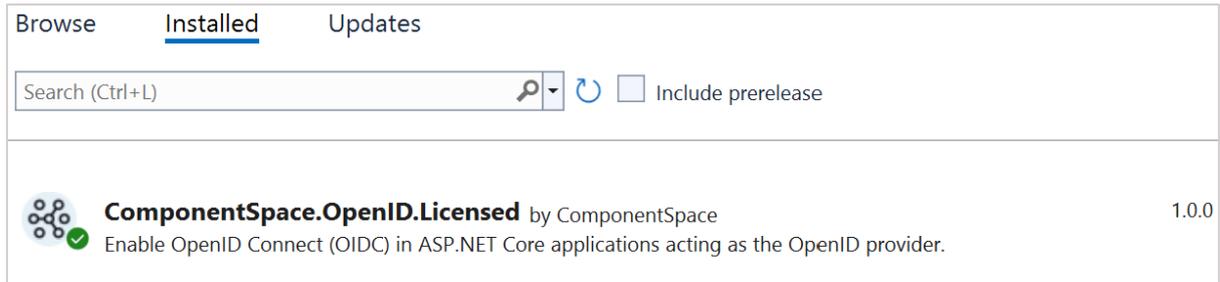
Figure 9 Licensed Version Installed

Your applications should build successfully without having to change the source code.

Be careful not to replace a licensed version of the product with an evaluation version from nuget.org.

Uninstallation

1. Uninstall the ComponentSpace.OpenID package from any ASP.NET Core web application projects.
2. Remove the ComponentSpace NuGet package source.
3. Delete the unzipped contents from the file system.