

NuGet Package Registry



- [Requirements](#)
- [Configuring the package registry](#)
- [Publish a package](#)
 - [Symbol Packages](#)
- [Install a package](#)
- [Supported commands](#)

Publish [NuGet](#) packages for your user or organization. The package registry supports the V2 and V3 API protocol and you can work with [NuGet Symbol Packages](#) too.

Requirements

To work with the NuGet package registry, you can use command-line interface tools as well as NuGet features in various IDEs like Visual Studio. More information about NuGet clients can be found in [the official documentation](#). The following examples use the `dotnet nuget` tool.

Configuring the package registry

To register the package registry you need to configure a new NuGet feed source:

```
dotnet nuget add source --name {source_name} --username {username} --password {password}
https://forgejo.example.com/api/packages/{owner}/nuget/index.json
```

Parameter	Description
<code>source_name</code>	The desired source name.
<code>username</code>	Your Forgejo username.
<code>password</code>	Your Forgejo password. If you are using 2FA or OAuth use a personal access token instead of the password.

Parameter	Description
<code>owner</code>	The owner of the package.

For example:

```
dotnet nuget add source --name forgejo --username testuser --password password123
https://forgejo.example.com/api/packages/testuser/nuget/index.json
```

You can add the source without credentials and use the `--api-key` parameter when publishing packages. In this case you need to provide a [personal access token](#).

Publish a package

Publish a package by running the following command:

```
dotnet nuget push --source {source_name} {package_file}
```

Parameter	Description
<code>source_name</code>	The desired source name.
<code>package_file</code>	Path to the package <code>.nupkg</code> file.

For example:

```
dotnet nuget push --source forgejo test_package.1.0.0.nupkg
```

You cannot publish a package if a package of the same name and version already exists. You must delete the existing package first.

Symbol Packages

The NuGet package registry has build support for a symbol server. The PDB files embedded in a symbol package (`.snupkg`) can get requested by clients. To do so, register the NuGet package registry as symbol source:

```
https://forgejo.example.com/api/packages/{owner}/nuget/symbols
```

Parameter	Description
<code>owner</code>	The owner of the package registry.

For example:

```
https://forgejo.example.com/api/packages/testuser/nuget/symbols
```

Install a package

To install a NuGet package from the package registry, execute the following command:

```
dotnet add package --source {source_name} --version {package_version} {package_name}
```

Parameter	Description
<code>source_name</code>	The desired source name.
<code>package_name</code>	The package name.
<code>package_version</code>	The package version.

For example:

```
dotnet add package --source forgejo --version 1.0.0 test_package
```

Supported commands

```
dotnet add
dotnet nuget push
dotnet nuget delete
```

	#3	
	cocobook-sync	 2026-10-28 16:38:51 UTC
	cocobook-sync	 2026-10-29 01:09:32 UTC