

CI/CD with Jenkins & .NET

AUGUST 3, 2018 NIDHI GUPTA 2 COMMENTS

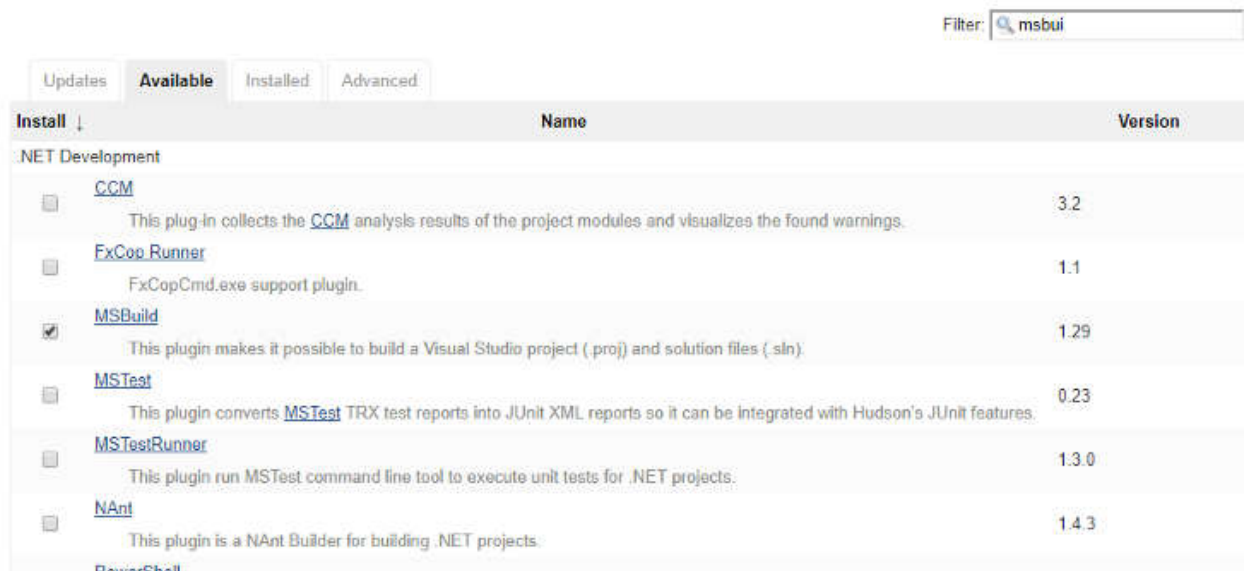
This blog will help you setup the CI/CD with Jenkins & .NET.

Tools Used:

- **Platform:** .NET 4.5.2
- **IDE:** Visual Studio 2015 (MSBuild files)
- **Application type:** Windows Service
- **NuGet:** Package Manager used for all references.
- **Source Control:** Git (github.com)

Install the MSBuild Plugin for Jenkins:

Manage Jenkins -> Manage Plugins -> Msbuild -> Install it



Configure MSBuild

Manage Jenkins—Global Tool Configuration ->

Path -> C:\Windows\Microsoft.NET\Framework\v4.0.30319 (This is by default present if .net framework is installed don your machine). Check the path and version and accordingly put in the jenkins configuration.

List of Gradle installations on this system

MSBuild

MSBuild installations

MSBuild
Name
Path to MSBuild

Specify the path to your msbuild executable.
It is usually located at C:\WINDOWS\Microsoft.NET\Framework\[version].
(from [Jenkins MSBuild Plugin](#))

Default parameters

☐ Install automatically

[Delete MSBuild](#)

[Add MSBuild](#)

List of MSBuild installations on this system

MSBuild is successfully configured.

Now we will create a job in Jenkins

1. Open Jenkins Url -> Create a free style project
2. In Source Control Management – Select Github -
><https://github.com/devops4solutions/jenkins-ci-dotnet.git>

General Source Code Management Build Triggers Build Environment Build Post-build Actions

Source Code Management

☐ None
☒ Git

Repositories

Repository URL <https://github.com/devops4solutions/jenkins-ci-dotnet.git>

Credentials [- none -](#) [Add](#)

[Advanced...](#)

[Add Repository](#)

Branches to build

Branch Specifier (blank for 'any') [*/master](#)

[Add Branch](#)

Repository browser [\(Auto\)](#)

Add a build Step -> Execute Windows Batch Command -> nuget restore src/MyWindowsService/

Build

Execute Windows batch command

Command

`nuget restore src/MyWindowsService/`

See [the list of available environment variables](#)

Advanced...

Add another build step -> Build Visual Studio project ->
src/MyWindowsService/MyWindowsService/Deploy-Windows-Service-Via-MSBuild.proj

Build a Visual Studio project or solution using MSBuild

MSBuild Version: MSBuild

MSBuild Build File: src/MyWindowsService/MyWindowsService/Deploy-Windows-Service-Via-MSBuild.proj

Command Line Arguments:

Pass build variables as properties: ☐

Do not use chcp command: ☐

Advanced...

Now save the project and trigger the build.

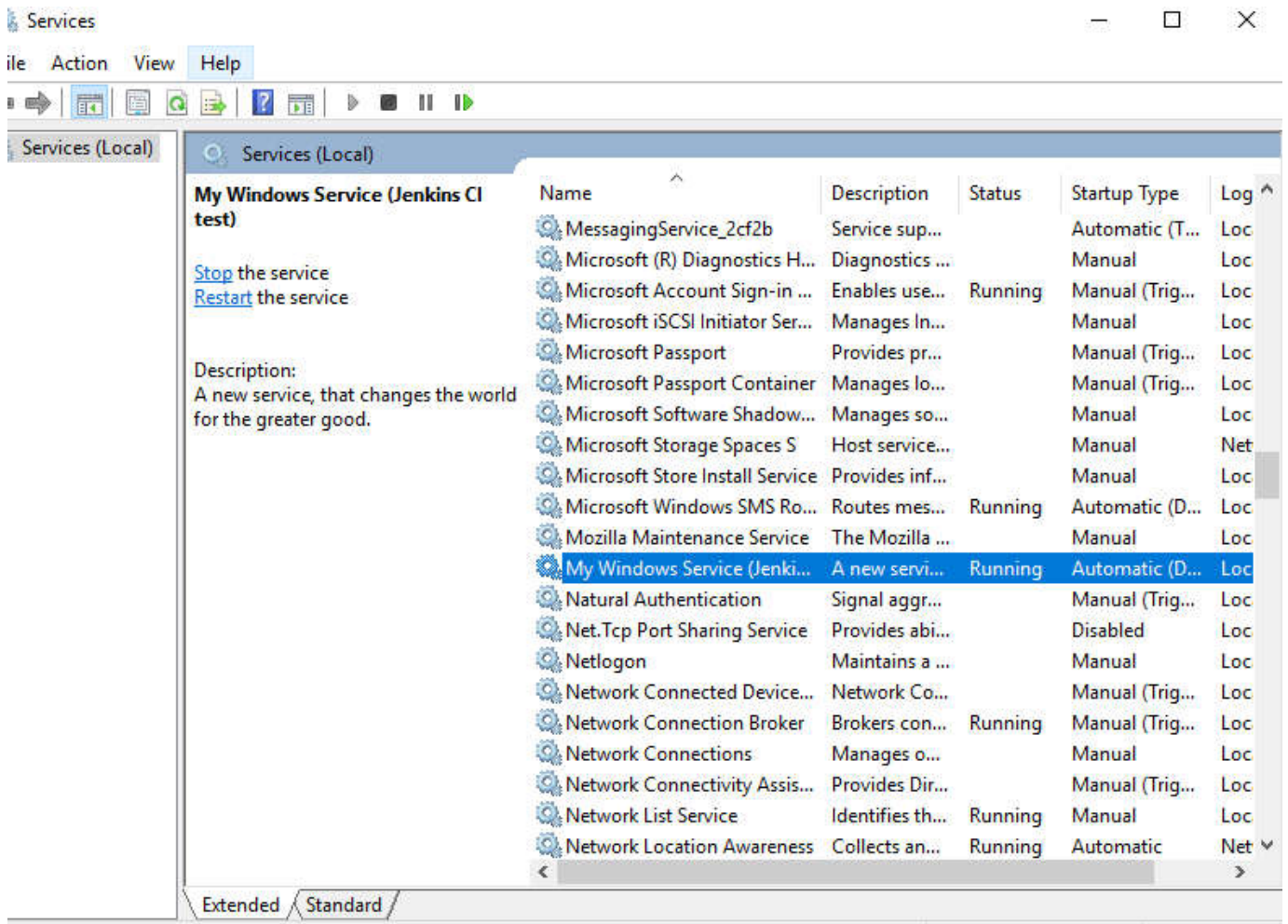
```
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
Waiting for MyWindowsService to start
MyWindowsService on is started
Done Building Project "C:\Program Files (x86)\Jenkins\workspace\CI_CD_Dotnet\src\MyWindowsService\MyWindowsService\Deploy-Windows-Service-Via-MSBuild.proj" (default targets).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:01:47.19
Finished: SUCCESS
```

Check the Results

Open Services -> check whether the services got created as shown in the screenshot



Congratulations, finally you have setup a CI/CD with Jenkins & .NET.