

ASP.NET Core Out Of Process Hosting Model

What is Out of Process Hosting in ASP.NET Core?

- ▶ In the case of the ASP.NET Core OutOfProcess Hosting Model, there are two web servers.
 - ▶ An internal webserver which is the Kestrel web Server
 - ▶ And an external web server which can be IIS, Apache, and Nginx.
- ▶ The most important point that you need to keep in mind is depending on how you are running your application with the OutOfProcess hosting model, the external web server may or may not be used.
- ▶ As we already discussed that the Kestrel web server is a cross-platform web server that is already embedded with your ASP.NET Core application. So if you are using the Out of Process Hosting model for your asp.net core application, then the Kestrel web server can be used.

How to Configure OutOfProcess Hosting

- ▶ We can configure the Out Of Process Hosting in two ways in ASP.NET Core.

- ▶ **Way1:**

- ▶ In this case, you just need to add the **<AspNetCoreHostingModel>** element to the applications project file with a value of **OutOfProcess** as shown below.

```
<Project Sdk="Microsoft.NET.Sdk.Web">
  <PropertyGroup>
    <TargetFramework>netcoreapp2.2</TargetFramework>

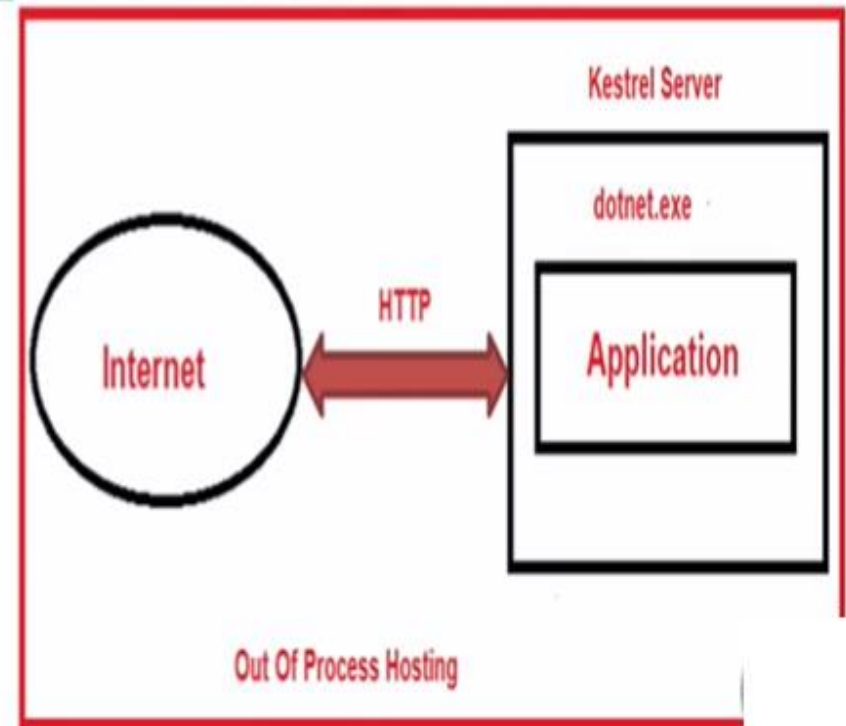
    <AspNetCoreHostingModel>OutOfProcess</AspNetCoreHostingModel>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="Microsoft.AspNetCore.App" />
    <PackageReference Include="Microsoft.AspNetCore.Razor.Design"
      Version="2.2.0" PrivateAssets="All" />
  </ItemGroup>
</Project>
```

- ▶ **Way2:**

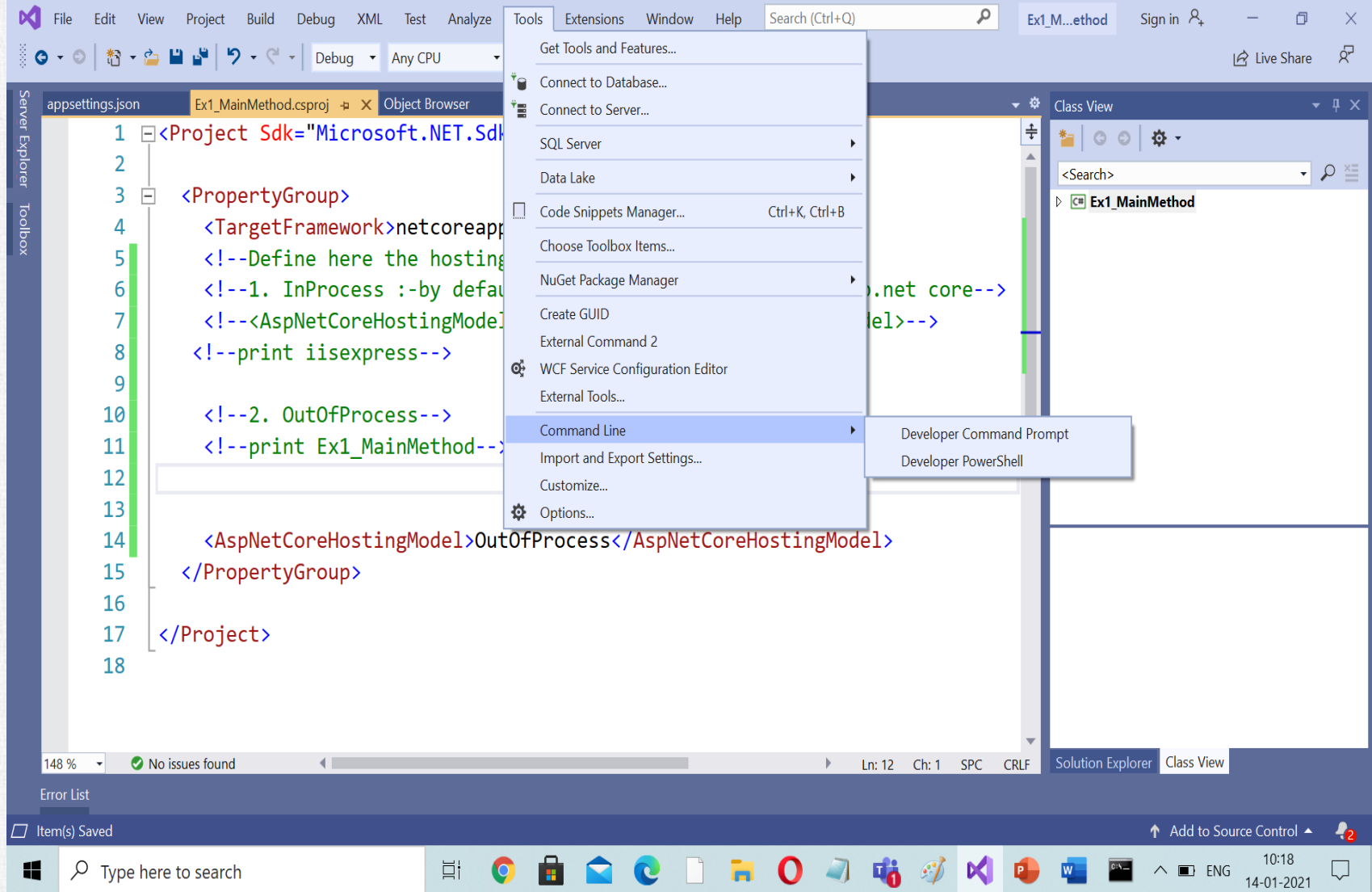
- ▶ The default hosting in ASP.NET Core is OutOfProcess Hosting. That means if you remove the **<AspNetCoreHostingModel>** element from the application's project file, then by default OutOfProcess hosting will be used.

How the Kestrel web server can be used?

- ▶ We can use the Kestrel Web Server as the internet-facing web server which will directly process the incoming HTTP requests.
- ▶ In this scenario, only the Kestrel Server is used and the other one i.e. external web server is not going to be used.
- ▶ So when we run the application using the .NET core CLI then Kestrel is the only web server that is going to be used to handle and process the incoming HTTP request as shown in the image.



Run using CLI:-



C:\Windows\system32\cmd.exe - dotnet run

```
*****  
** Visual Studio 2019 Developer Command Prompt v16.8.2  
** Copyright (c) 2020 Microsoft Corporation  
*****
```

C:\Users\Lenovo\Desktop\Asp.net_Core\Examples\Ex1_MainMethod>dotnet run

```
info: Microsoft.Hosting.Lifetime[0]  
      Now listening on: http://localhost:5000  
info: Microsoft.Hosting.Lifetime[0]  
      Application started. Press Ctrl+C to shut down.  
info: Microsoft.Hosting.Lifetime[0]  
      Hosting environment: Development  
info: Microsoft.Hosting.Lifetime[0]  
      Content root path: C:\Users\Lenovo\Desktop\Asp.net_Core\Examples\Ex1_MainMethod
```



Type here to search



ENG

10:18
14-01-2021



- If we are not writing hosting model in proj file run the application print the process name iisexpress.
- `<AspNetCoreHostingModel>InProcess</AspNetCoreHostingModel>`
- Output:-iisexpress
- If we run the application using CLI it will run the application in Kestrel server
- `<AspNetCoreHostingModel>OutOfProcess</AspNetCoreHostingModel>`
- Output is Application name:-
- By default OutOfProcess model is used in CLI

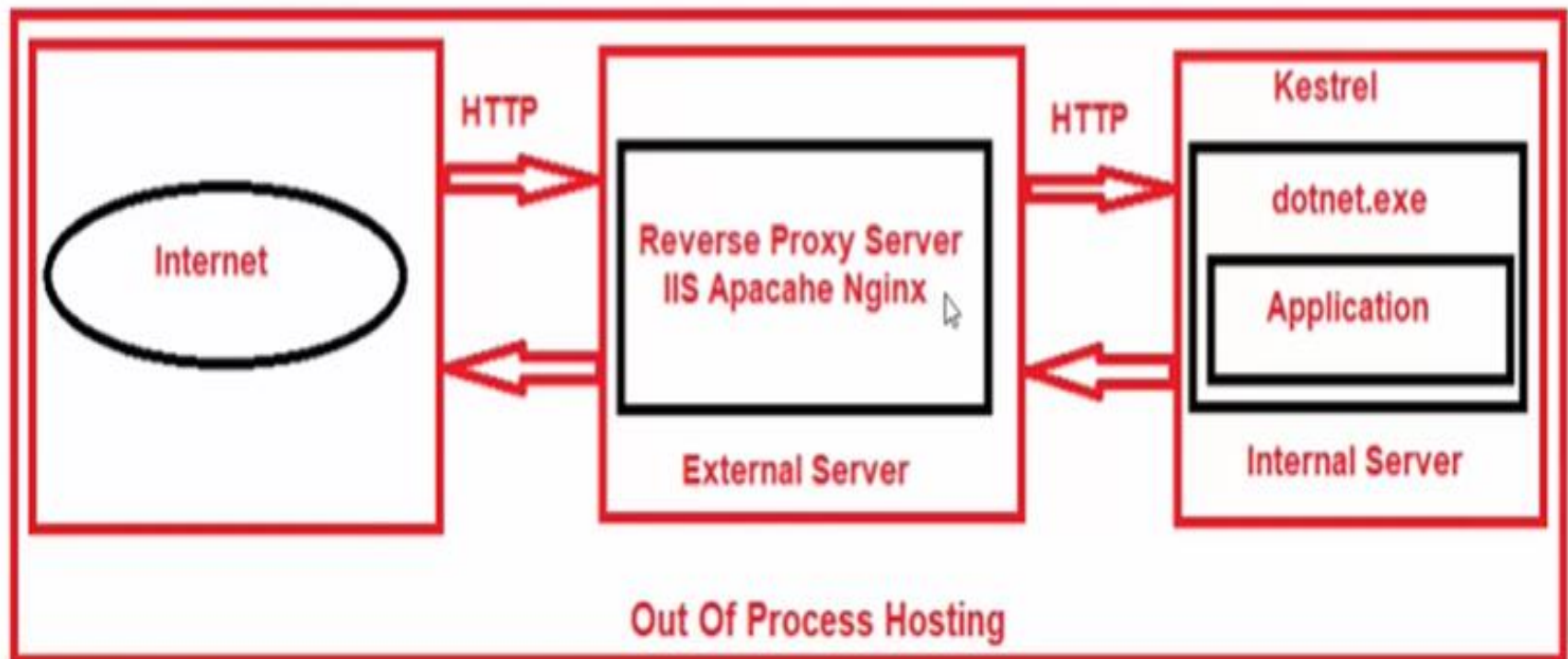
How the Kestrel web server can be used?

- ▶ The Kestrel Web Server can also be used with the combination of a reverse proxy server such as IIS, Apache or Nginx.

why do we need a reverse proxy server?

- ▶ This is because the reverse proxy server provides an additional layer of configuration and security which is not available with the Kestrel Server.
- ▶ It also maintains the load balancing. So it is a good choice to use Kestrel Server along with a reverse proxy server.
- ▶ So when we use Kestrel Server along with a reverse proxy server, then the reverse proxy server will receive the incoming HTTP requests from the client and then forwards that request to the Kestrel server for processing.
- ▶ Once the Kestrel Server process that request, then it sends the response back to the reverse proxy server which then sends the response back to the requested client over the internet.

Reverse Proxy Server



Can we run an asp.net core application without using the built-in kestrel web server?

- ▶ YES. When we use the InProcess Hosting model, then the application is hosted inside the IIS worker process i.e. w3wp.exe in the case of IIS and iisexpress.exe in the case of IIS Express.
- ▶ That means the Kestrel Web Server is not used with the InProcess hosting model.



Thanks