Версия .NET SDK 8, Runtime 8   
Чтобы загрузить и развернуть приложение ASP.NET на другом сервере, вам нужно выполнить несколько шагов по установке и настройке серверной среды. Вот что нужно установить и настроить:

**1. Установка .NET на сервере**

Прежде чем развернуть ASP.NET приложение на сервере, нужно убедиться, что на сервере установлен .NET. Для этого следуйте этим шагам:

1. **Загрузите .NET SDK или Runtime:**
   * Перейдите на сайт [официальной страницы .NET](https://dotnet.microsoft.com/download) и выберите нужную версию .NET (например, .NET 6 или .NET 7).
   * Если вы хотите, чтобы приложение работало в режиме "self-contained" (независим от установленного .NET), вам нужно скачать и установить .NET Runtime. Если сервер уже имеет установленный .NET, можно использовать **Framework-dependent** режим.
2. **Установите .NET:**
   * Запустите установщик и следуйте инструкциям на экране. Вы можете установить как SDK (если хотите разрабатывать на сервере), так и Runtime (если сервер только будет запускать ваше приложение).
3. После установки .NET перезагрузите сервер, если это требуется.

**2. Установка IIS (если не установлен)**

ASP.NET приложение можно запускать через **IIS (Internet Information Services)** — веб-сервер от Microsoft. Чтобы использовать IIS, выполните следующие шаги:

1. Откройте **Диспетчер серверов** (Server Manager).
2. Перейдите в раздел **"Добавить роли и компоненты"**.
3. В мастере установки выберите роль **"Web Server (IIS)"**.
4. Пройдите по шагам мастера, чтобы установить IIS.
5. Убедитесь, что установлены компоненты для поддержки **ASP.NET**:
   * В мастере установок IIS проверьте, чтобы была выбрана роль **"ASP.NET"**.

После установки IIS перезагрузите сервер.

**3. Настройка IIS для работы с вашим приложением**

1. **Настройка веб-сайта в IIS:**
   * Откройте **Менеджер IIS**.
   * В панели **Сайты** выберите **Добавить сайт**.
   * Укажите имя сайта (например, MyAspNetApp), путь к папке, где находится ваше опубликованное приложение, и порт, на котором будет работать сайт (например, порт 8000).
   * Убедитесь, что **используется приложение ASP.NET**.
2. **Настройка файла web.config:**
   * В папке с вашим приложением должен быть файл web.config. Он необходим для того, чтобы IIS знал, как обрабатывать ваше приложение.

Пример конфигурации для web.config:

xml

КопироватьРедактировать

<configuration>

<system.webServer>

<aspNetCore processPath="dotnet" arguments="MyApp.dll" stdoutLogEnabled="false" stdoutLogFile=".\logs\stdout" />

</system.webServer>

</configuration>

**4. Публикация и развертывание приложения**

1. Опубликуйте ваше ASP.NET приложение через Visual Studio или командную строку в папку на сервере.
2. Скопируйте файлы в папку на сервере, где IIS будет искать ваш сайт.

Для публикации через Visual Studio:

* Нажмите правой кнопкой мыши на проект в **Solution Explorer**.
* Выберите **Publish**.
* В качестве цели выберите **Folder** и укажите папку на сервере, например, C:\inetpub\wwwroot\MyAspNetApp.

1. После этого IIS должен распознавать ваше приложение и начинать его запуск.

**5. Открытие порта 8000 в брандмауэре Windows**

Если вы хотите, чтобы приложение работало на порту 8000, убедитесь, что этот порт открыт в брандмауэре:

1. Откройте **Windows Firewall**.
2. Перейдите в **Дополнительные параметры** -> **Правила для входящих подключений**.
3. Нажмите **Создать правило** и выберите **Порт**.
4. Укажите порт 8000 и разрешите подключение.

**6. Тестирование приложения**

После того как вы настроили сервер, вы можете проверить доступность приложения, открыв браузер и перейдя по адресу:

cpp

КопироватьРедактировать

http://<server-ip>:8000

Если все настроено правильно, ваше приложение должно запуститься.

**Резюме:**

* Установите **.NET SDK/Runtime** на сервере.
* Установите и настройте **IIS** для обработки вашего ASP.NET приложения.
* Настройте сайт в IIS и перенесите файлы приложения на сервер.
* Откройте порт 8000 в брандмауэре Windows.
* Проверьте доступность приложения через браузер.