This a note for remind me how to publish ASP.NET Core website to AWS via Visual studio.
PS: My Visual studio version is Enterprise 2017. Hope it also helps you, good luck.

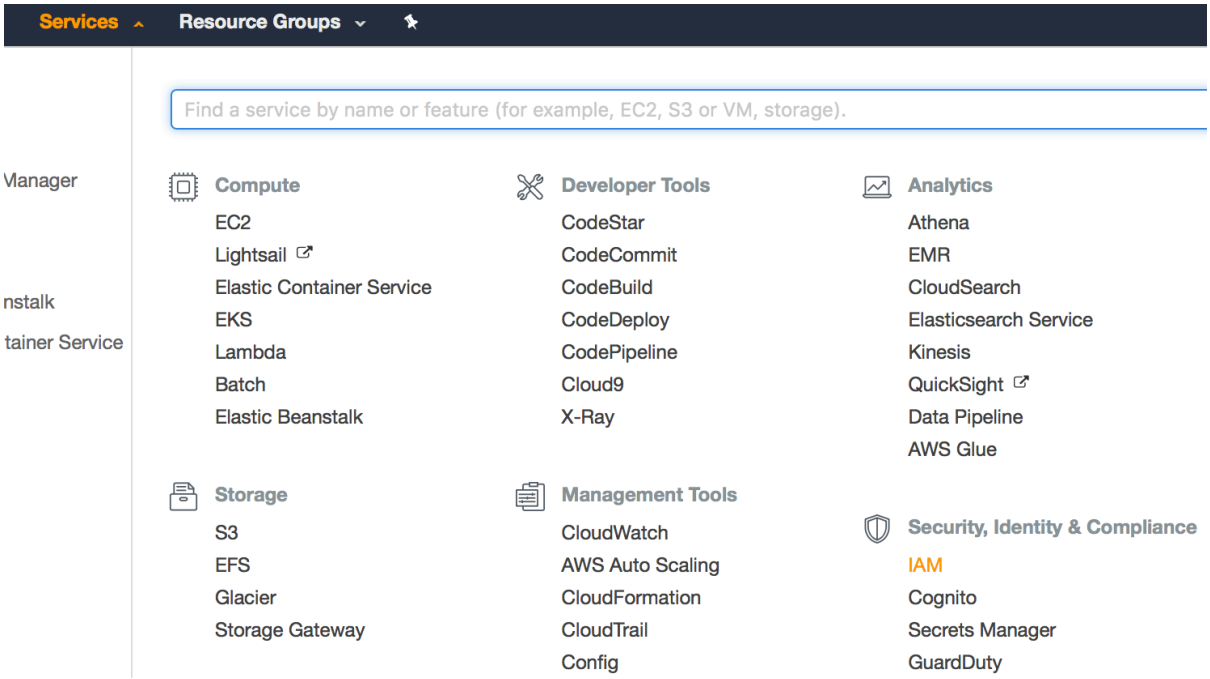
Step 1: AWS setting

If you already create a user and add access policy for that user, please go to step 2.

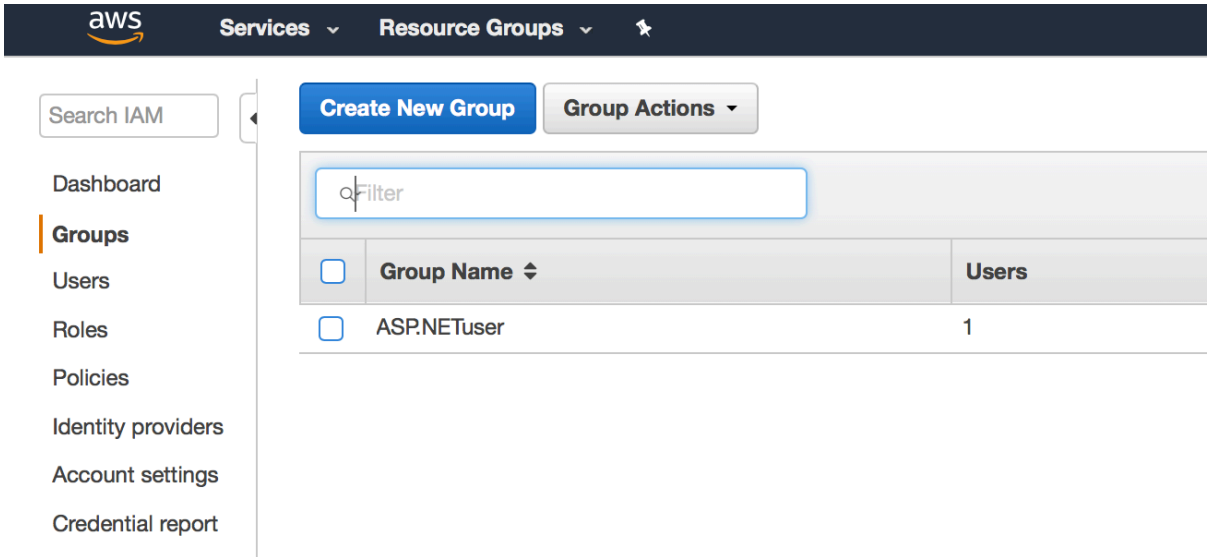
If you don't have an AWS account, click the link to create an AWS account:
https://portal.aws.amazon.com/billing/signup#/start

Please note that, for authorize your AWS account, you have to bind a valid bank account to your AWS account which can use online and it has at least $1.

After your AWS account was authorized, login to your account 'Service' -> 'Security, Identity & Compliance' -> 'IAM'

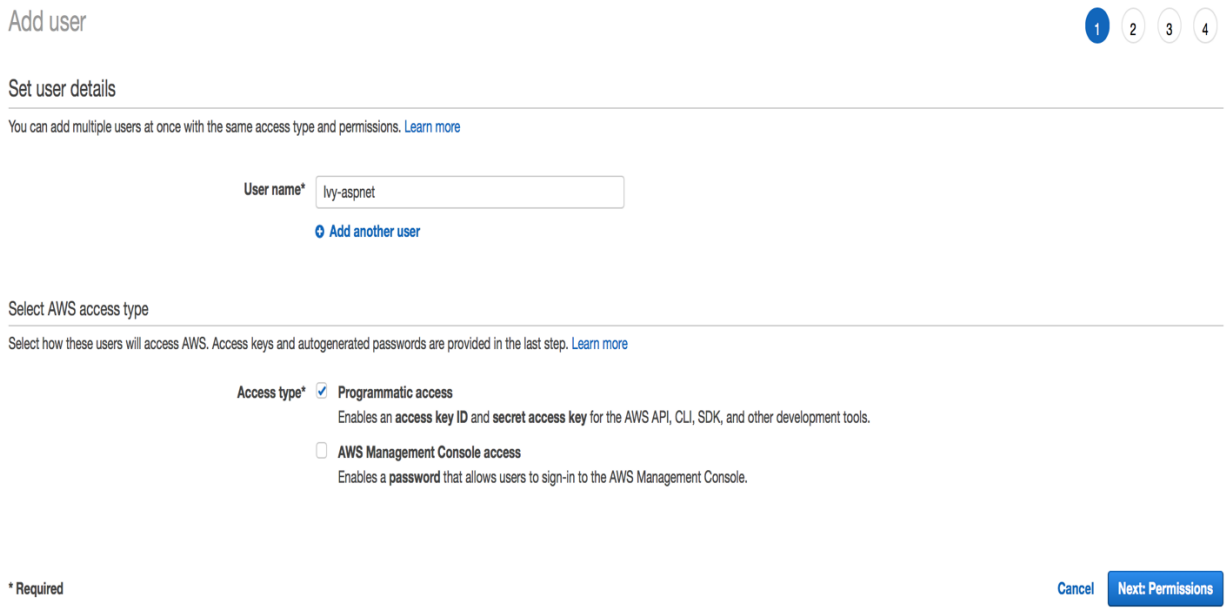


In left hand side, select 'Groups' -> 'Create New Group' (if you don't want to put user in a group, you can create new user and attach policy directly.)

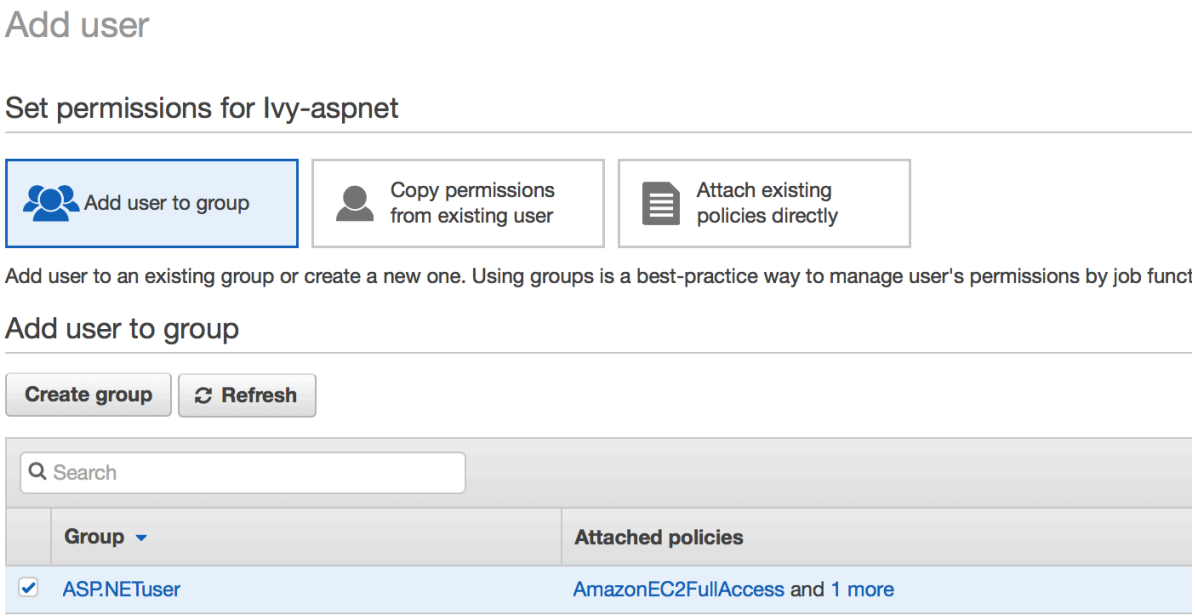


Input your group name, click 'Next' to attach policy, select 'AmazonEC2FullAccess' & 'AWSElasticBeanstalkFullAccess'

After you created group, select 'User' in the left-hand side bar, click 'Add New User', input user name, the access type chose 'Programmatic access'



Add user to the group, it's ok if you don't have a group, but don't forget to attach access policy for your user, just the same as attach access policy to a group.



After you add user successfully, you will have 'Access key ID' & 'Secret access key', remember this, you will need this for you AWS setting in visual studio.

Add user

1 2 3 4

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: https://638982746235.signin.aws.amazon.com/console

Download .csv

User	Access key ID	Secret access key
Ivy-asp.net		

Close

Step 2: Visual studio setting

For configure the Visual Studio environment, you have to install ‘AWSSDK.EC2’ & ‘AWSSKD’

There two ways to install ‘AWSSDK.EC2’:

The first way is install it via ‘Visual studio’ -> ‘Tools’ -> ‘NuGet package Manager’, you won’t have ‘AWS explorer’

```
PM> Install-Package AWSSDK.EC2
```

```
PM> Install-Package AWSSDK
```

Another way is to download ‘MSI Installer’. The installer will include the AWS Toolkit for Visual Studio and AWS Tools for PowerShell, you will get a ‘AWS explorer’ in your VS.

Secure | https://aws.amazon.com/sdk-for-net/

aws Contact Sales Products Solutions Pricing Getting Started More English My Account Create an AWS Accour

RESOURCES

AWS SDK for .NET

RELATED LINKS

Documentation

Tools

Get Started for Free

Create Free Account

AWS SDK for .NET

Get started quickly using AWS with the AWS SDK for .NET. The SDK helps take the complexity out of coding by providing .NET APIs for AWS services including Amazon S3, Amazon EC2, Amazon DynamoDB and more. The SDK can be downloaded from NuGet or installed using the MSI package, which also includes the AWS Toolkit for Microsoft Visual Studio 2013 and 2015 editions and the AWS Tools for Windows PowerShell.

The AWS SDK for .NET is now distributed as multiple service-specific packages on NuGet. Example package names include AWSSDK.EC2, AWSSDK.S3, and AWSSDK.DynamoDB. Each of these depends on the AWSSDK.Core, which will automatically be installed in your project if you reference any of the service packages in the NuGet Package Manager.

Getting Started » Developer Guide » API Reference » Developer Blog »

Install

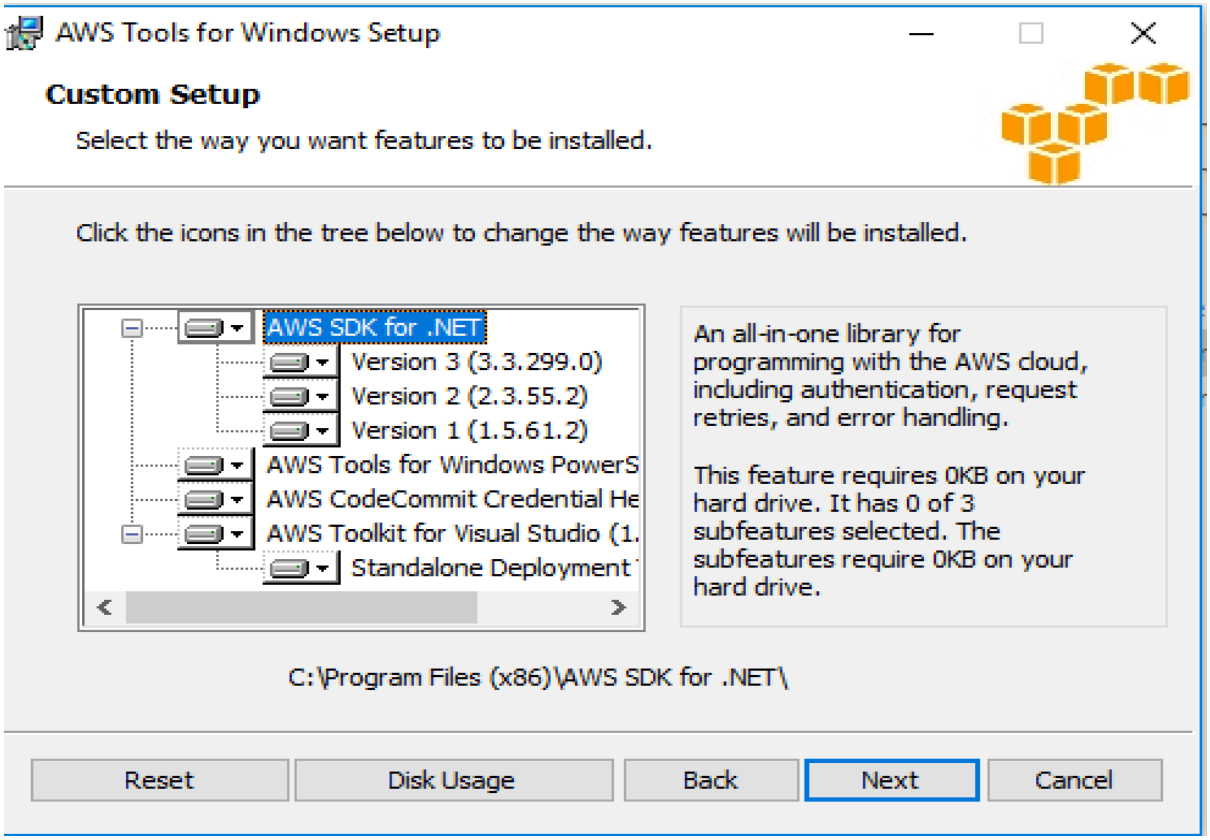
Install from NuGet »

NuGet is the recommended way to install AWS SDK for .NET packages. NuGet Package Manager will install the correct assemblies for your project type - .NET 3.5, 4.5, or Portable Class Library.

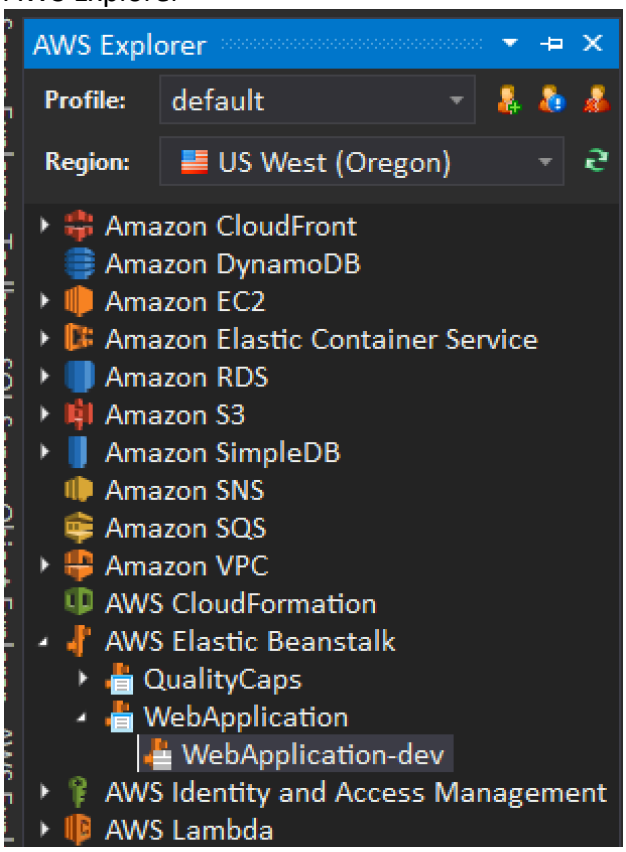
Download MSI Installer »

If you prefer to use the MSI installer, you can download it by clicking the above button. The installer also includes the AWS Toolkit for Visual Studio and AWS Tools for Windows PowerShell. The installer includes assemblies for .NET 3.5 and 4.5.

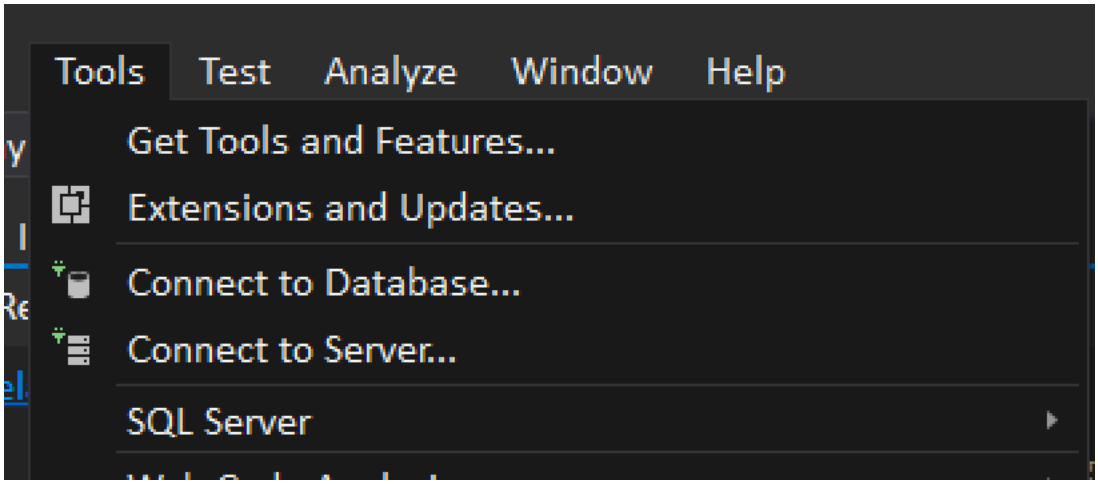
Installing AWS Tools …



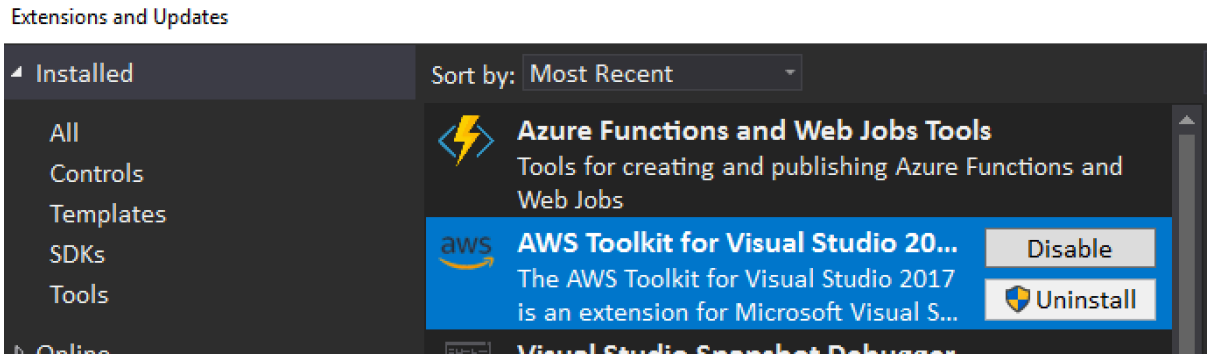
AWS Explorer



Install ‘AWS Toolkit’ via visual studio, ‘Tools’ -> ‘Extensions and Updates’

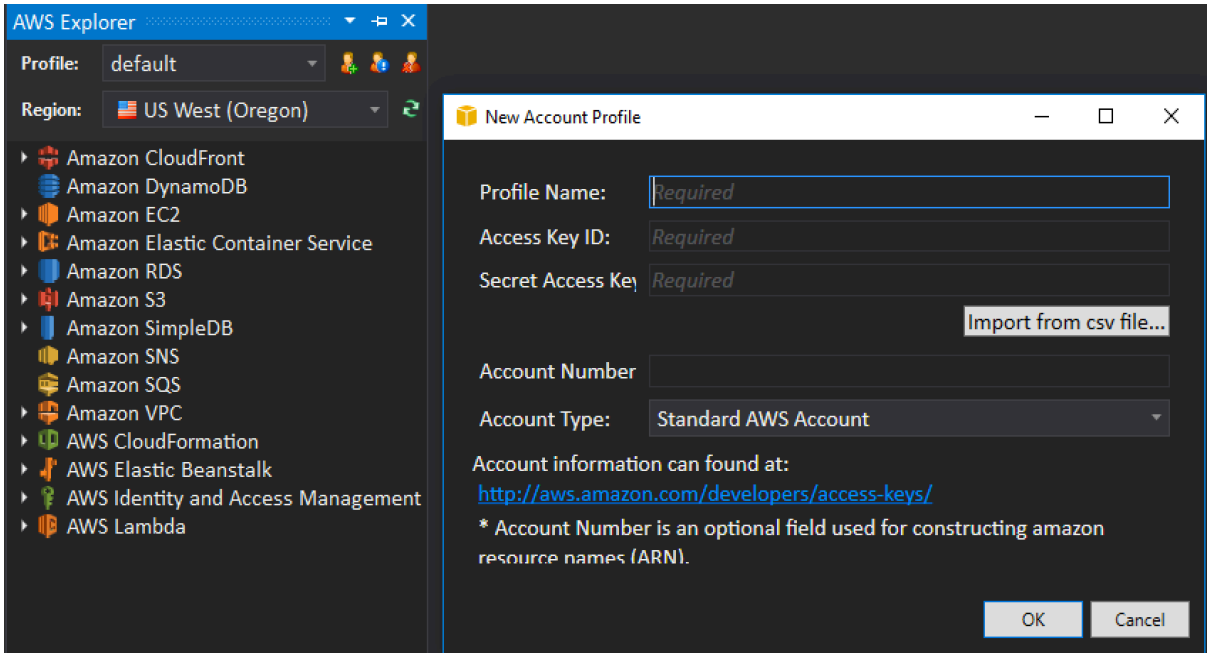


Once you install it, you see it in your ‘Extensions and Updates’ -> ‘Installed’ page



All tools are installed.

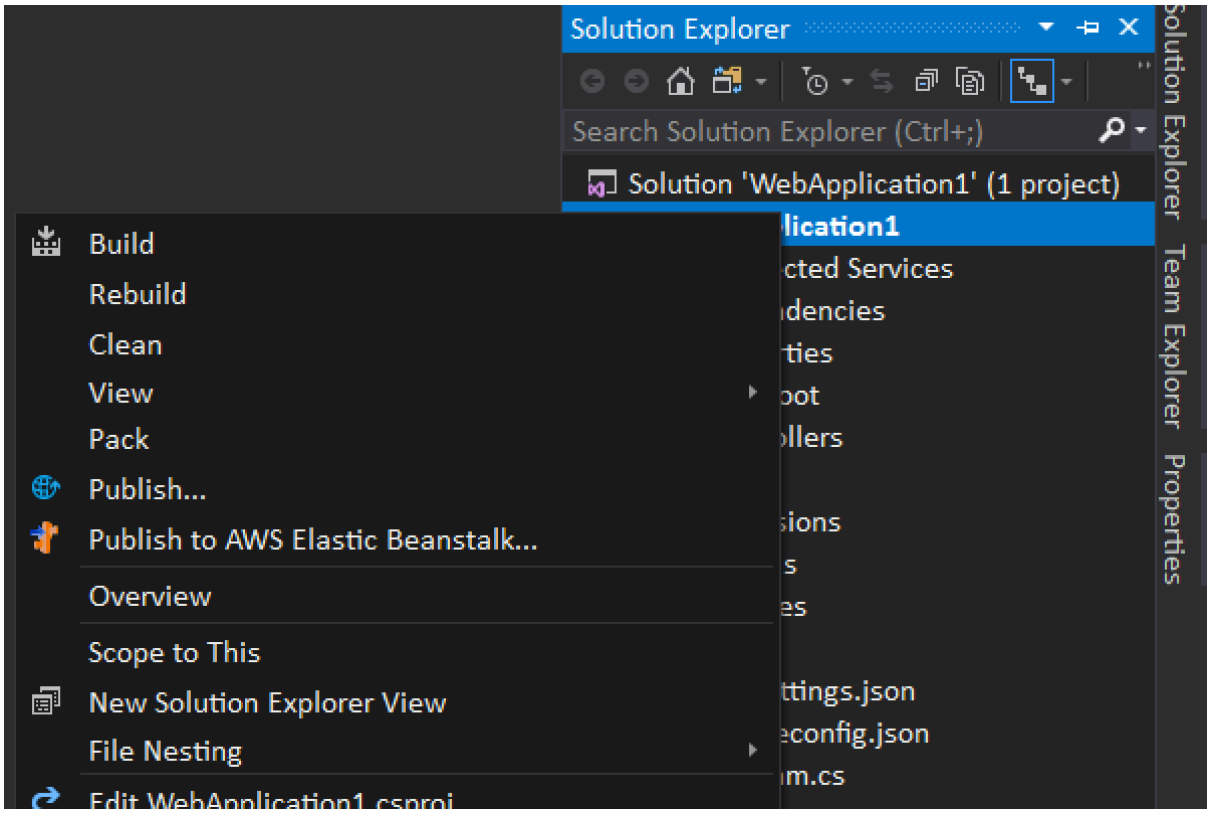
Add profile to your ‘AWS Explorer’, click ‘+’, ‘New Account Profile’ window will show. Input a profile name, the have ‘Access key ID’ & ‘Secret access key’ is the keys you got from your ‘AWS’-> ‘IAM’-> ‘User’



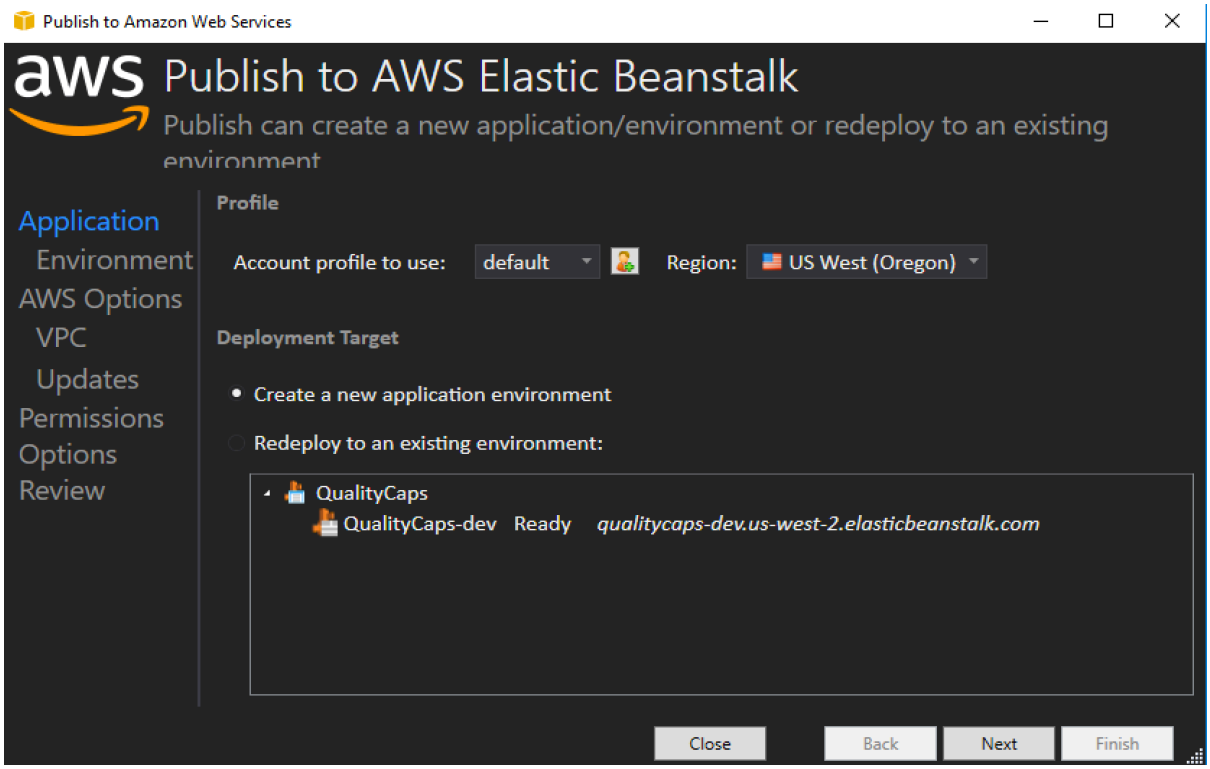
Everything is done, let's start to publish our website to AWS via visual studio.

Step 3: Publish website to AWS via visual studio

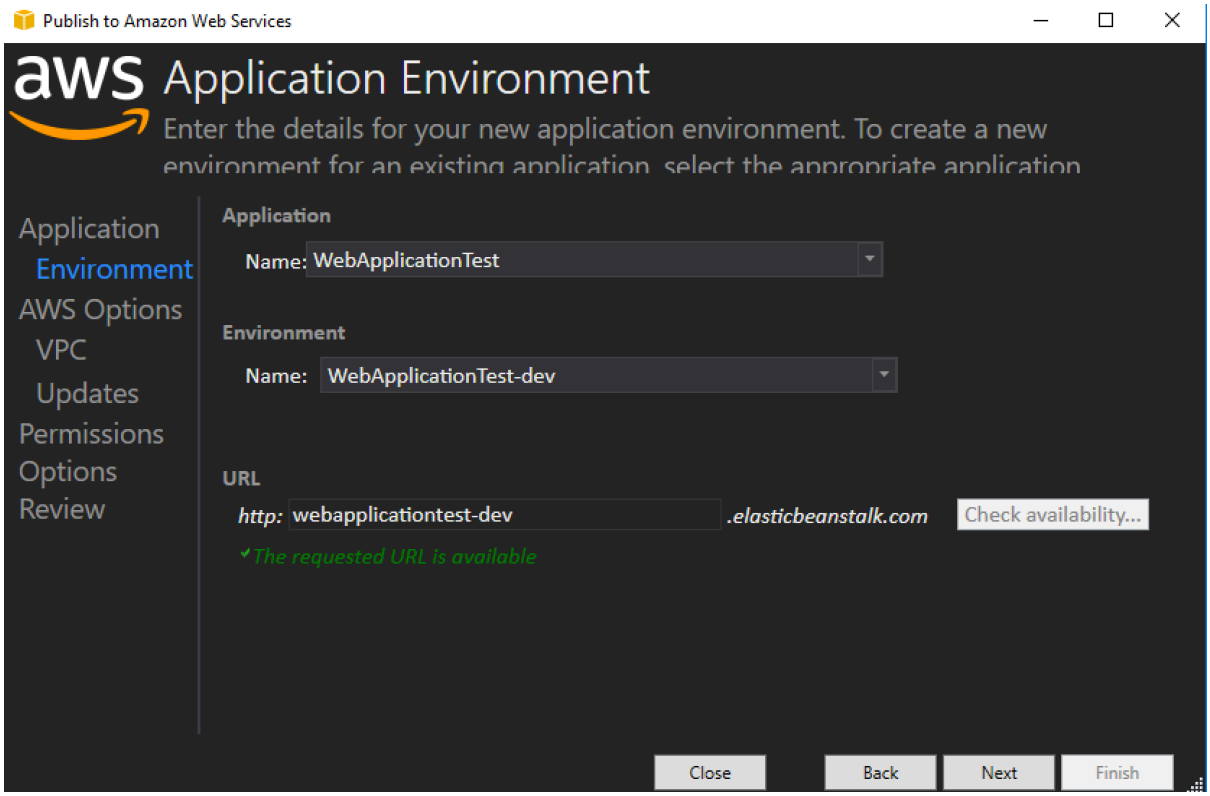
Right click on your project, select 'Publish to AWS Elastic Beanstalk'…



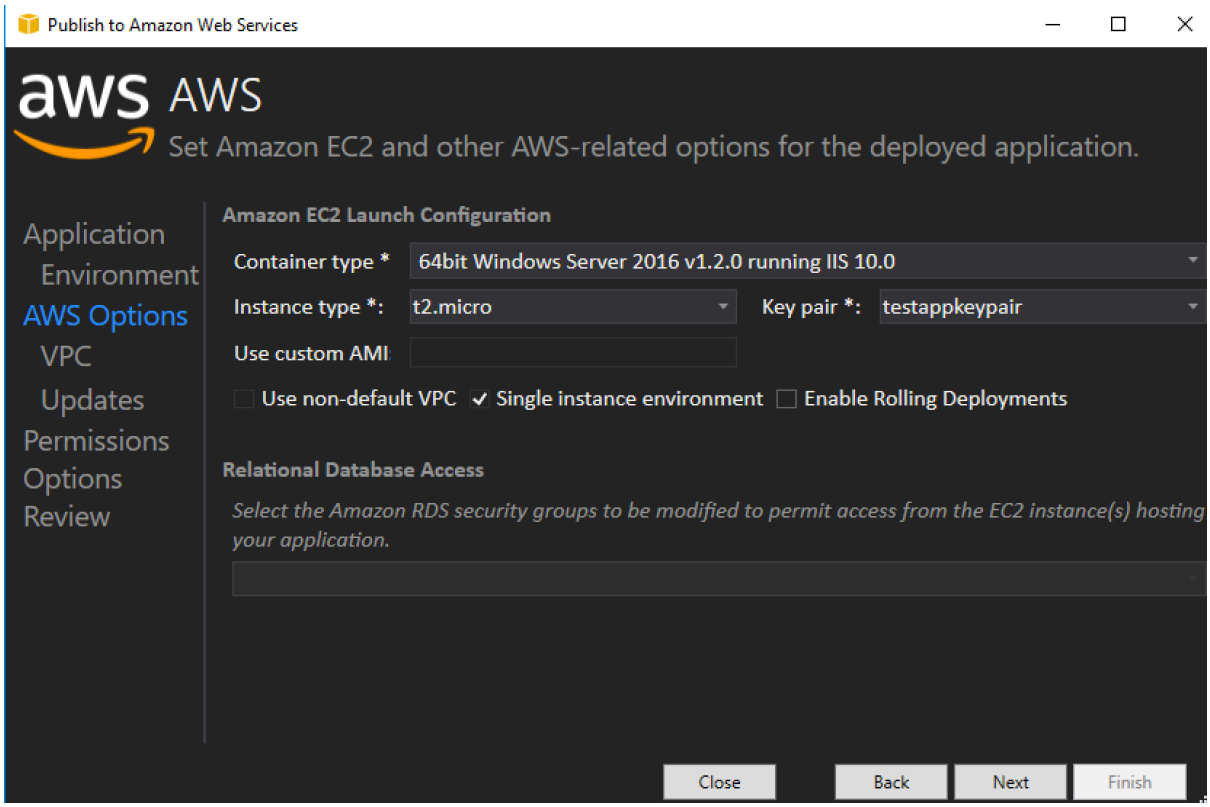
You can choose different account profile, for me, I just have one profile which connect to AWS account. You can choose 'Redeploy to and existing environment' or 'new environment'



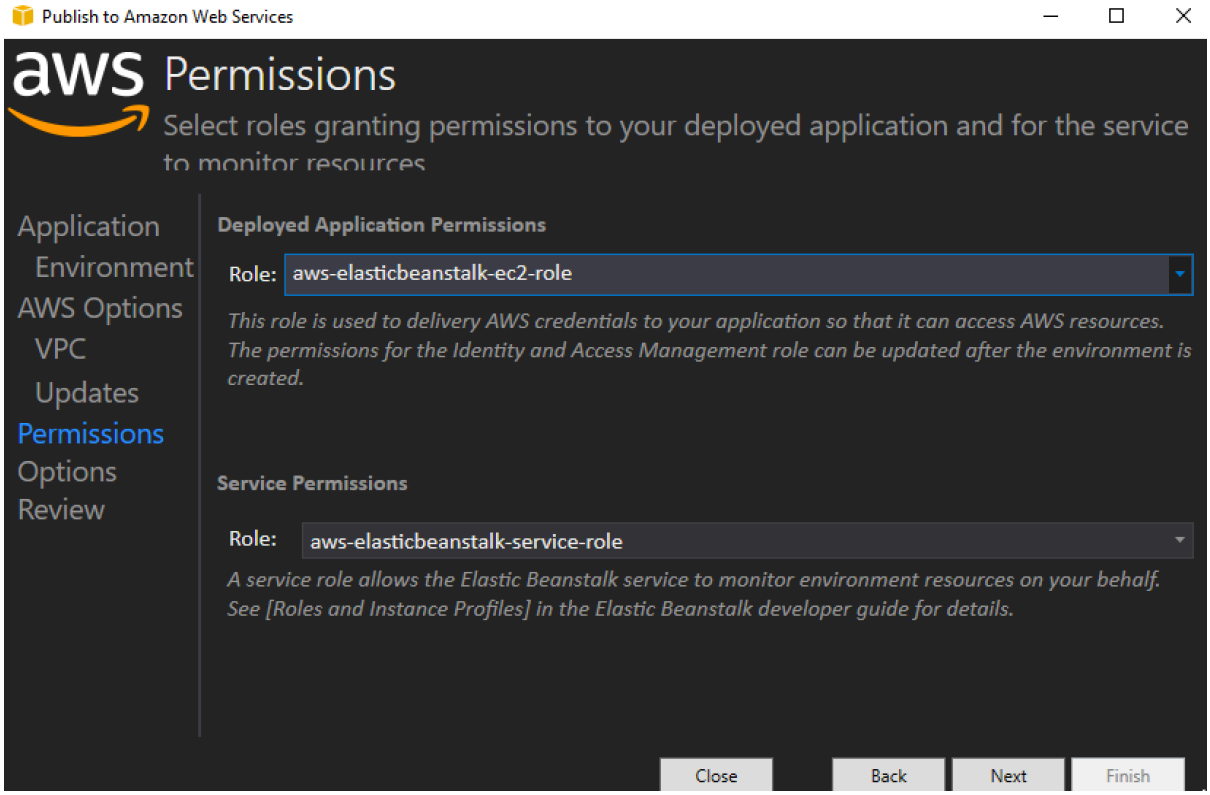
Check the URL is available or not



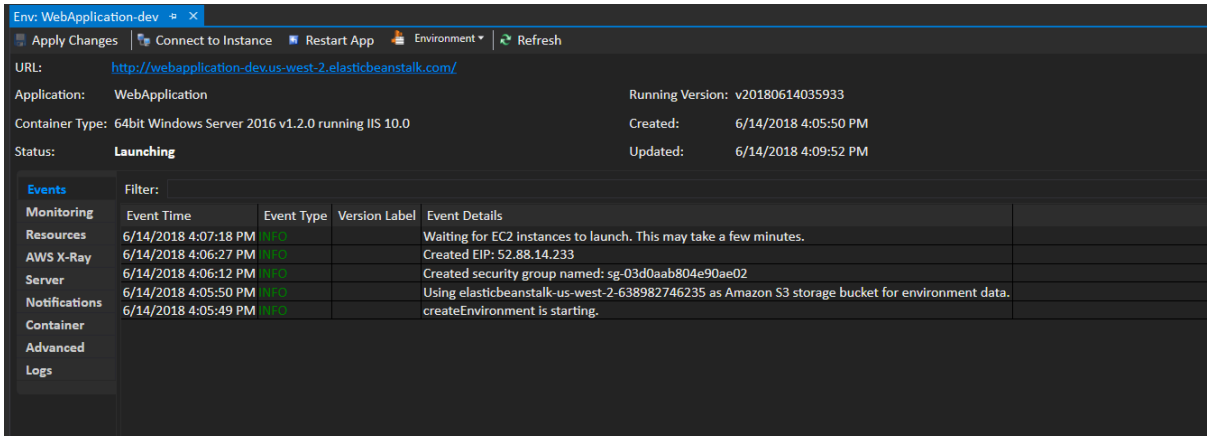
I create a new key pair, just click dropdown arrow -> ‘Create a new key pair’



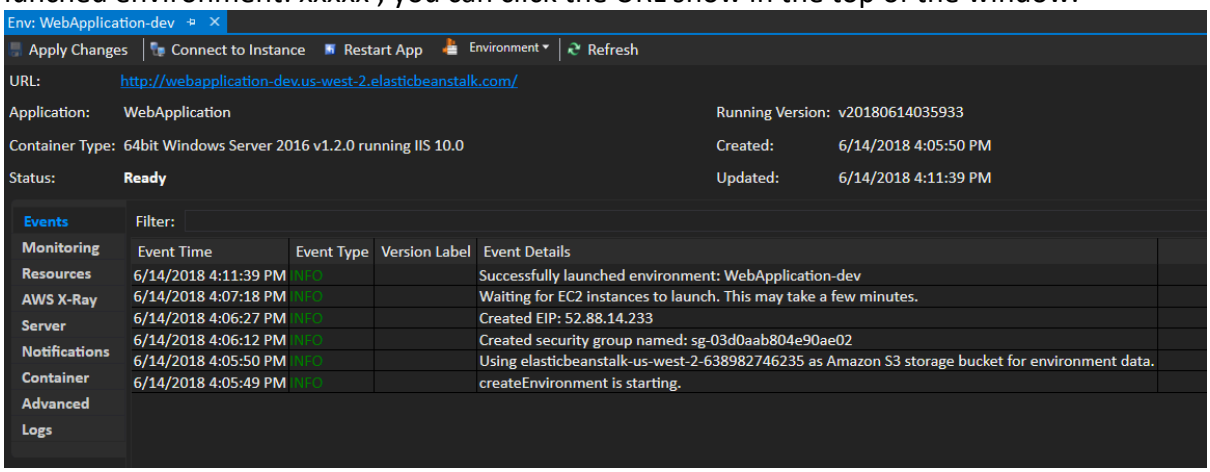
The Deployed Application Permissions will auto fill-up for you



click ‘Next’ -> ‘Deploy’ button, it will start to publish your project, you can see this info from ‘AWS Explorer’



Once the project was done, the ‘AWS Explorer’ window will show ‘INFO Successfully lunched environment: xxxxx’, you can click the URL show in the top of the window.



My sample project was published successfully. So far, everything is done, enjoy your journey at C# .NET Core and AWS…