Introduction

Azure Key Vault is a tool for securely storing and accessing secrets. A secret is anything that you want to tightly control access to, such as API keys, passwords, or certificates. A vault is a logical group of secrets.

You can refer to the below Microsoft document for more details.

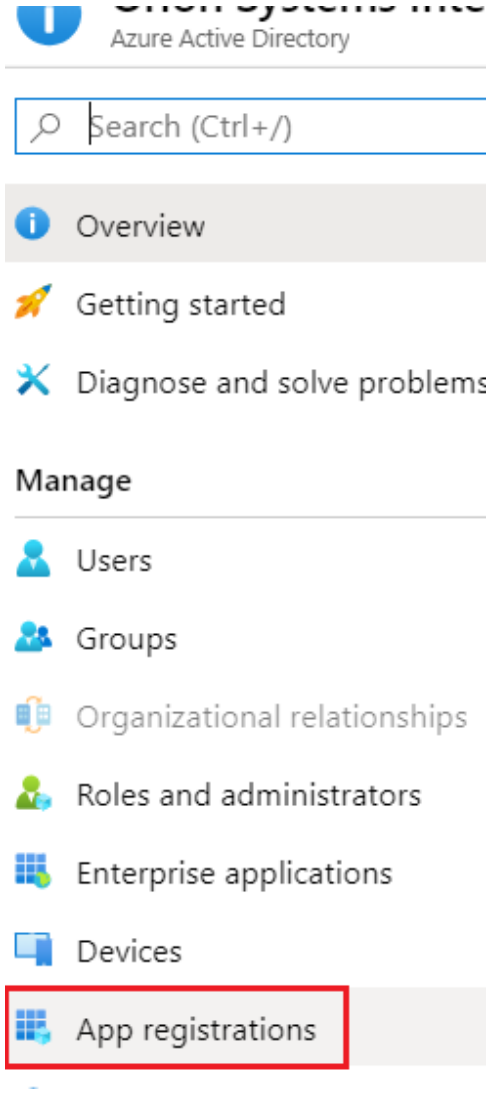
https://docs.microsoft.com/en-us/azure/key-vault/basic-concepts

Prerequisites

Azure portal access, Visual Studio 2019 or Visual Studio Code
Along with Azure Key Vault, we need an Azure App Registration in Azure Active Directory to ac
Key Vault secrets. Let's create App registration first.

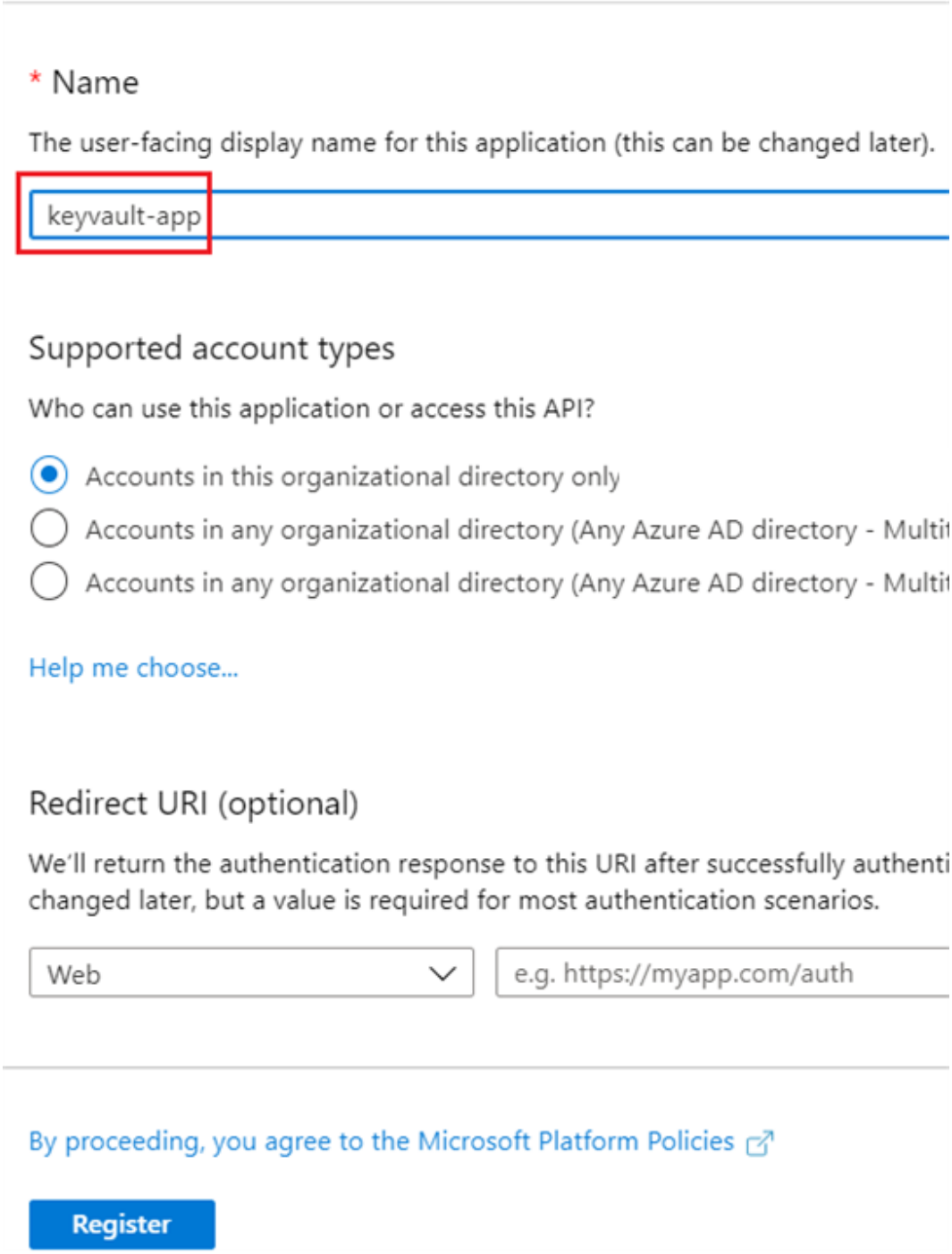
Create App registration in Azure Active Directory

Open Azure portal and click Azure Active Directory blade and click “App registrations” tab.

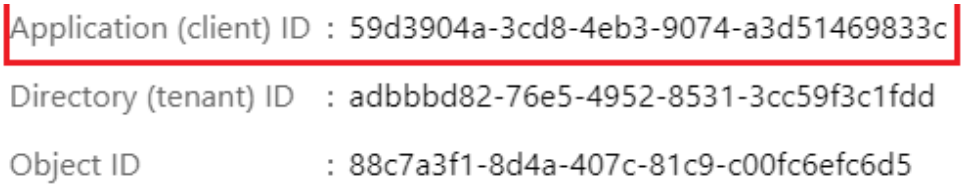


Click “New registration” tab to create new app registration.

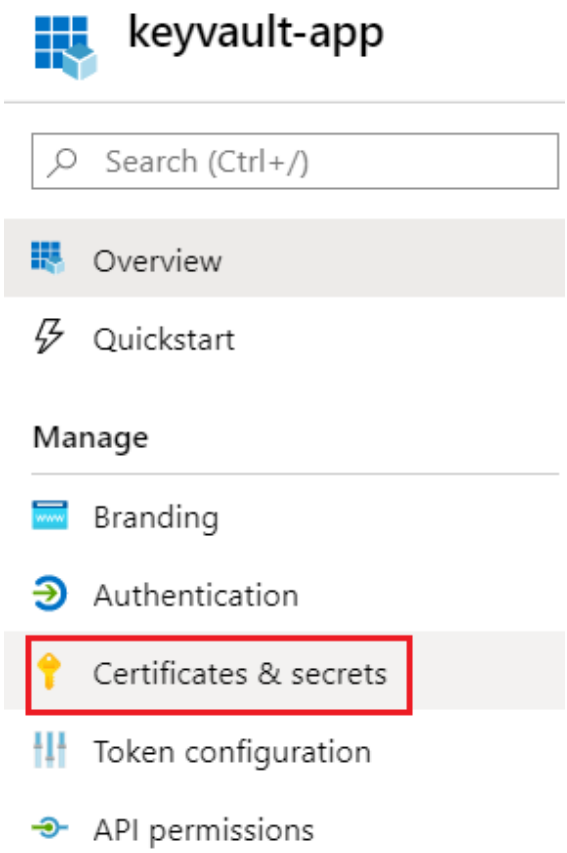
We can give a valid name to app registration and click Register button to proceed.



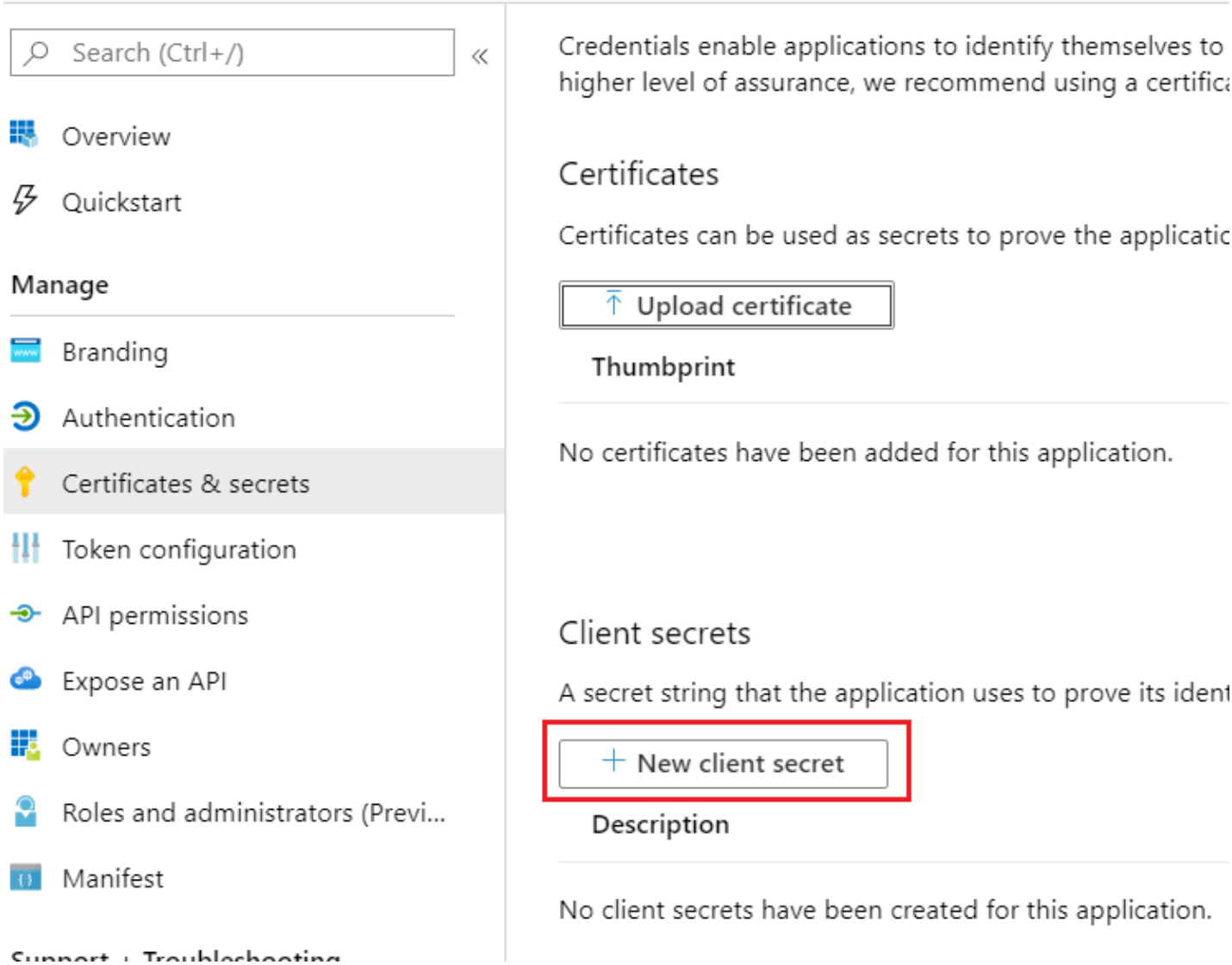
Please copy the Application ID (Client ID) to any secure place. We will use this ID in our Web API application later.



We can create a client secret in this app registration. Click “Certificates & secrets” tab.



Click “New client secret” button to create a new client secret.

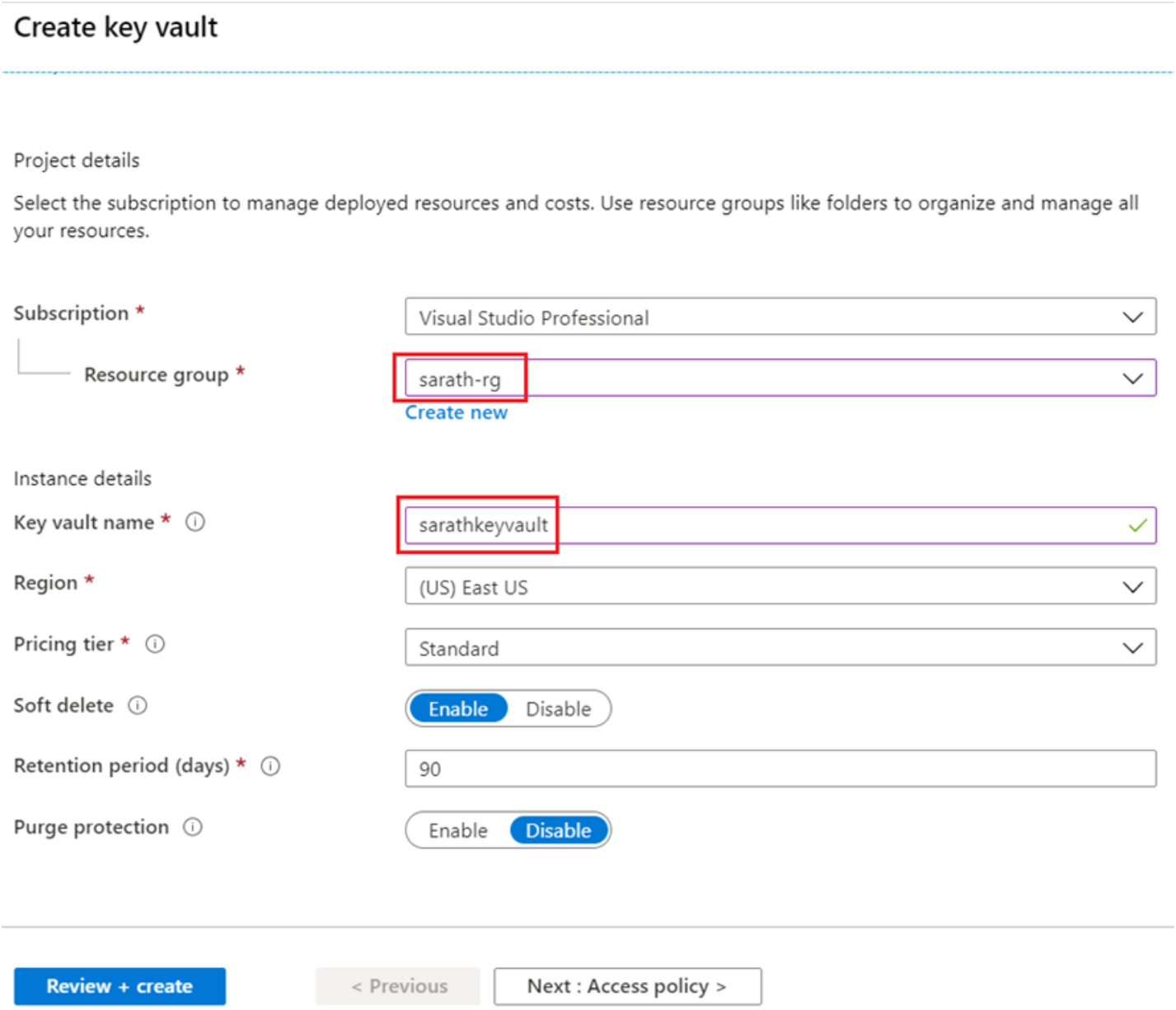


We can give any description and create client secret.

Please copy the above secret key and keep it in any secure place. We will use this value also in Web API applications.
We have successfully completed the app registration part and copied the required values like client id and client secret value. We can create the Azure Key Vault now

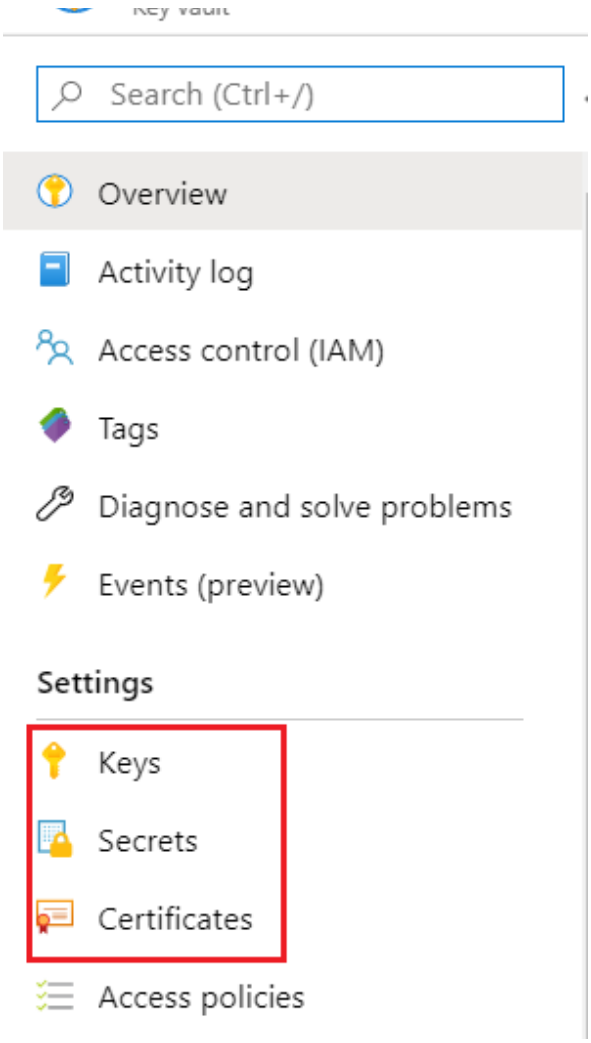
Create Azure Key Vault and Secret Value

Click “create” button



We can choose existing resource group or create new resource group. Please give a valid name to key vault. Also choose appropriate region. I have kept all other fields as default. If you want to modify, you can do it carefully.

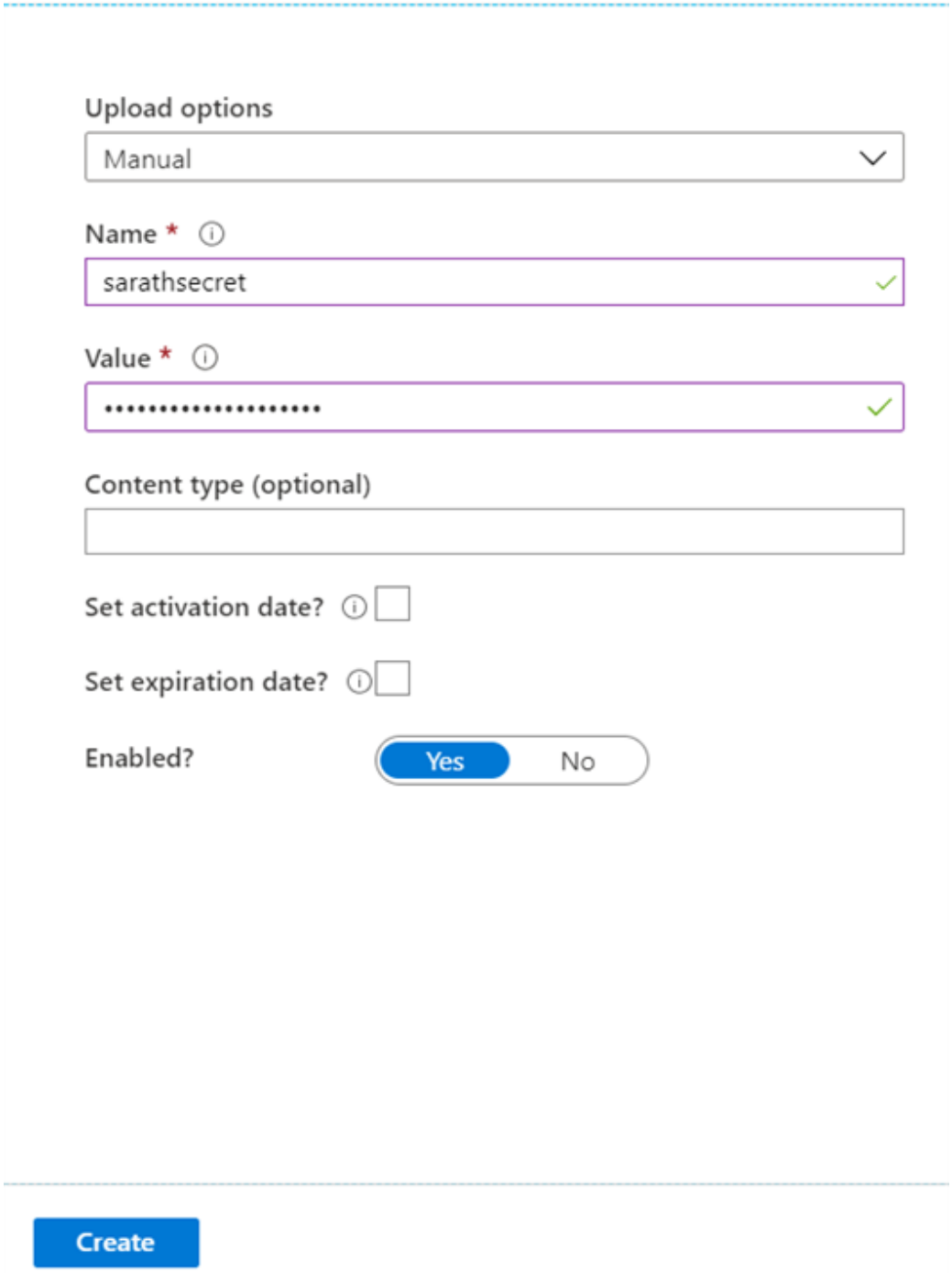
Please click “Review + create” button. Your Key Vault will be deployed in a few moments.



There are three types of Key vaults available. Keys, Secrets, and Certificates. In this article, we will see Secrets only.

We can click “Secrets” to create a new secret key and value pair.

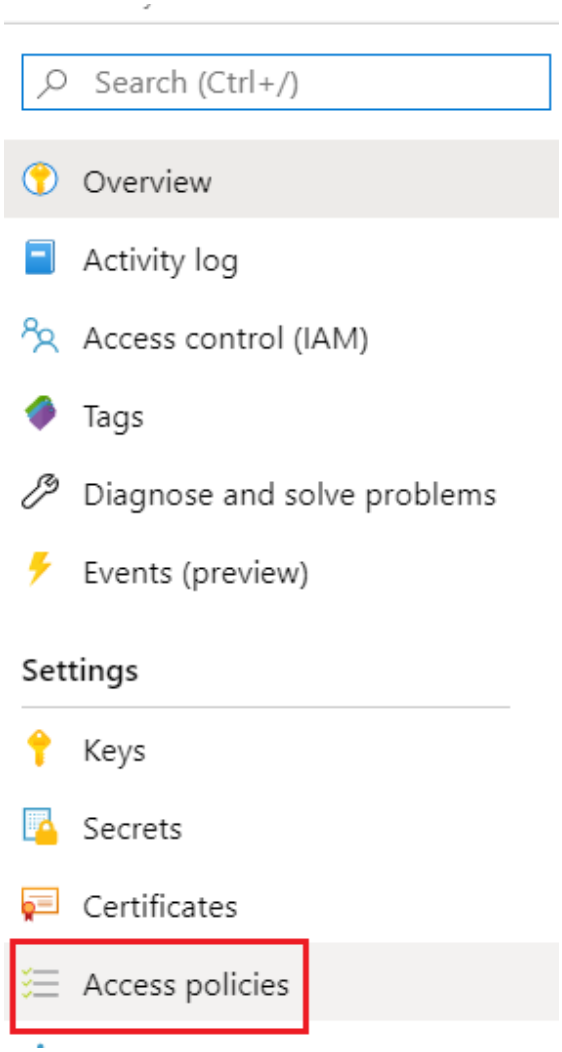
Click “Generate/Import” button to create new secret pair.



We can give a name and value to the secret.

Click “Create” button to create secret value pair.

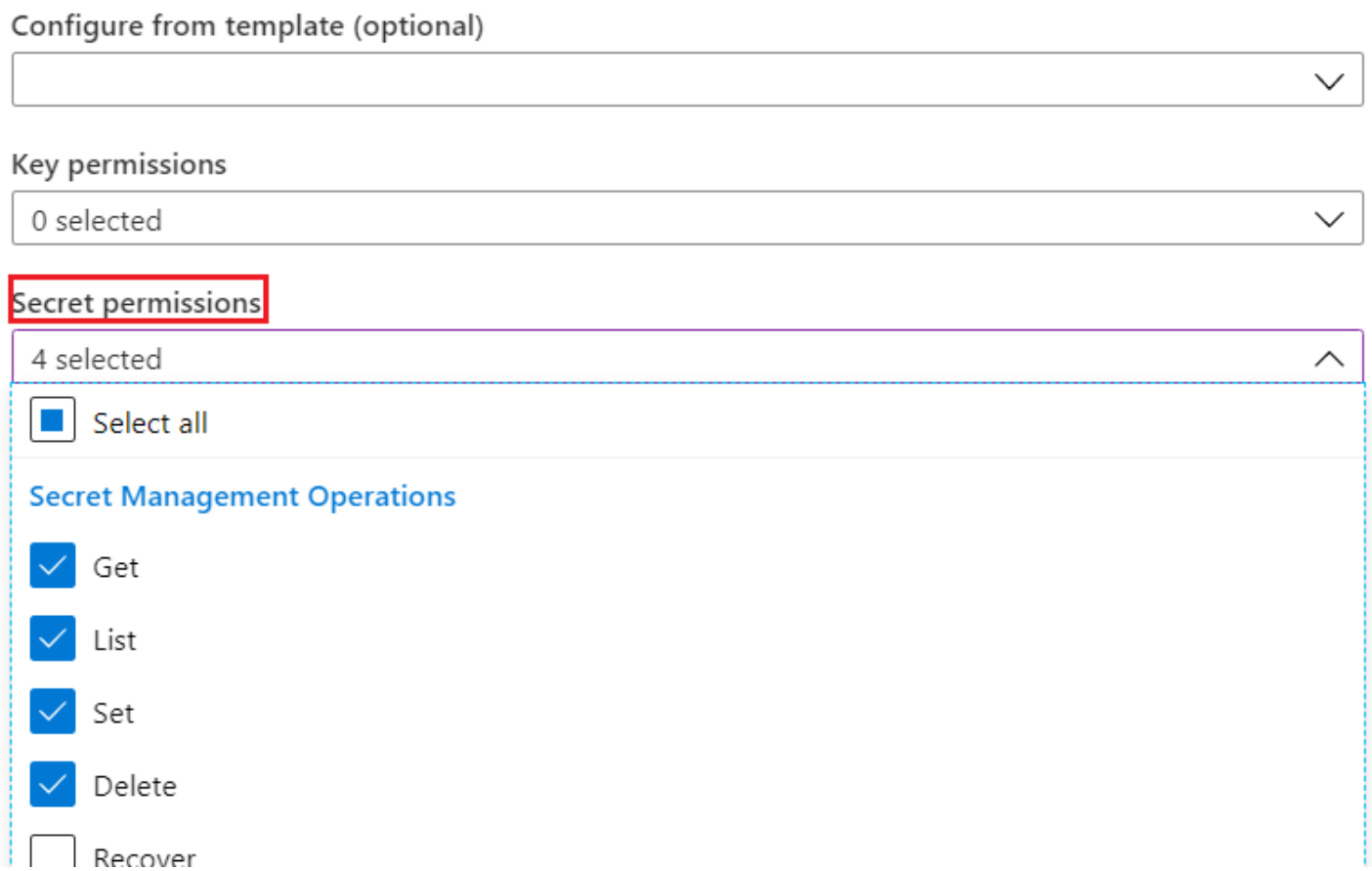
We can grant access policies of this Key Vault to app registration, which we have created already.



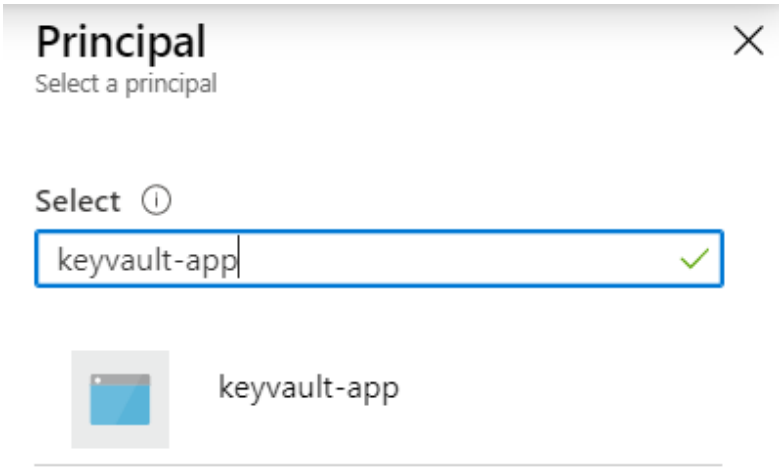
Click “Access policies” tab to proceed.

Click “+ Add Access Policy”

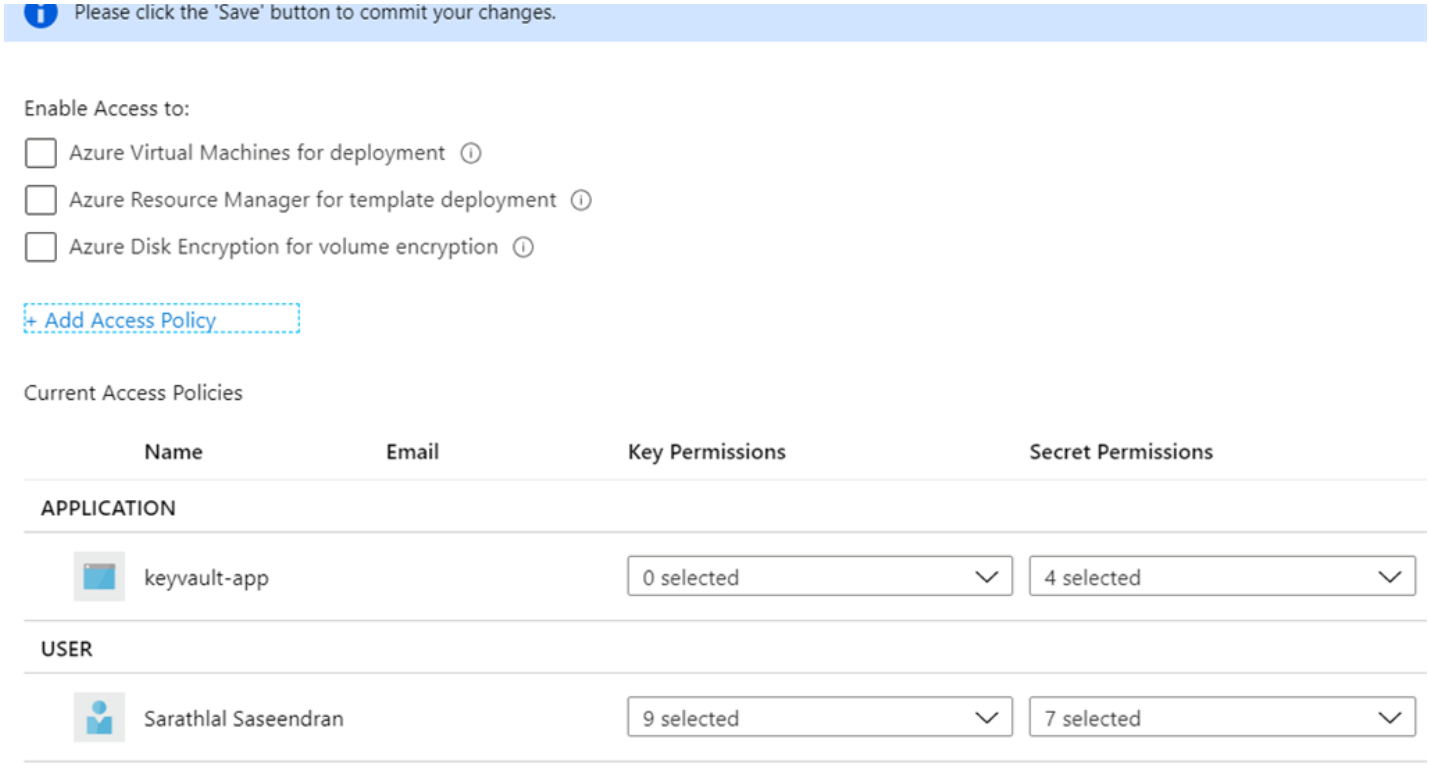
Choose secret permissions and choose Get, List, Set, and Delete.



Select principal and search for our app registration name. We have already created an app registration. Select it and click “Add” button.



We can see the selected app registration with secret permissions from Key Vault. We can save the permissions.



We have successfully created Azure Key Vault and Secret key value pairs. We can create a Web application and consume these details and get secret value from Key Vault.

Create Web API Core application in Visual Studio 2019

We can create a simple Web API application with ASP.NET Core template.

Modify the appsettings.json with the below values.

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },

  "KeyVault": {
    "Vault": "sarathkeyvault",
    "ClientId": "59d3904a-3cd8-4eb3-9074-a3d51469833c",
    "ClientSecret": "O?ozhtVAk=DKf=D[cruXEqU1rtDf3I61"
  },

  "AllowedHosts": "*"
}
```

We can install “Microsoft.Extensions.Configuration.AzureKeyVault” NuGet package to the project.

We can modify the “CreateHostBuilder “method in Program.cs file.

Program.cs

```
using Microsoft.AspNetCore.Hosting;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.Hosting;

namespace AzureKeyVaultSecret
{
    public class Program
    {
        public static void Main(string[] args)
        {
            CreateHostBuilder(args).Build().Run();
        }

        public static IHostBuilder CreateHostBuilder(string[] args) =>
           Host.CreateDefaultBuilder(args)
            .ConfigureAppConfiguration((context, config) =>
            {

                var root = config.Build();
                config.AddAzureKeyVault($"https://{root["KeyVault:Vault"]}.v
            })
            .ConfigureWebHostDefaults(webBuilder =>
            {
                webBuilder.UseStartup<Startup>();
```

```
}
```

We can create a new API controller “ValuesController” under Controllers folder.

Modify the default code with the below code.

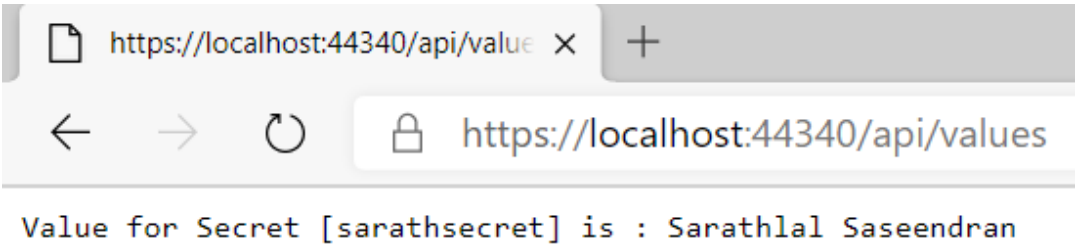
ValuesController.cs

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Configuration;

namespace AzureKeyVaultSecret.Controllers
{
    [Route("api/[controller]")]
    public class ValuesController : Controller
    {
        private readonly IConfiguration _configuration;

        public ValuesController(IConfiguration configuration)
        {
            _configuration = configuration;
        }

        [HttpGet]
        public string Get()
        {
            var value = _configuration["sarathsecret"];
            return "Value for Secret [sarathsecret] is : " + value;
        }
    }
}
```

We can run the application and execute the below end point.

https://localhost:44340/api/values

You will get the below value in the screen.



We have successfully retrieved the value for Key Vault Secret into the Web API application.

Conclusion

In this post, we have created an app registration and also created a client secret for app registration. We have created a Key Vault with Secret and granted access permissions to app registration. Later we have created a ASP.NET Core Web API and fetched the secret value from Key Vault using Client Id and Client secret key.

ASP.NET Core Web API | Azure | Azure App Registration | Azure Key Vault | Key Vault Secret

Next Recommended Reading

Your Secrets Are Safe With Key Vault In ASP.NET Core Web App On Azure

Sarathlal Saseendran TOP 500

A passionate human being loves to learn new things always.

https://www.linkedin.com/in/sarathlal-saseendran/

113	5.3m	3
10	18	

Type your comment here and press Enter Key (Minimum 10 characters)

Follow Comments

Rajeev Chib
2056 15 0
Jun 24, 2021
0 0 Reply

Hi Sarathlal, Nice article.

Rajeev Chib
2056 15 0
Jun 24, 2021
0 0 Reply

Hi, When I run the solution I am getting an exception, "No such Host is known", can anyone help me on this?

Pratham Majithia
2063 8 0
Oct 26, 2020
0 0 Reply

Hi, Very nice article. Now i understand what is client id and secret key using to access key-vault. Is there any other option here to access key-vault without client Id and secret key? In microservices solution there will be multiple applications but have to use only one azure app configurations and one key-vault, then how to handle?

Preetam T
2054 17 0
Jul 14, 2020
0 0 Reply

I hope you delete all of your Key Vault settings and secrets now that you exposed them in this post. Also, your Azure Key Vault client id and secrets should not be in your app settings.json file for proper security. You should have that as part of your user secrets.

Joseph Guadagno
912 1.6k 0
Apr 29, 2020
1 2 Reply

I have deleted all my Key Vault settings and secrets which was simply used for demonstration purpose. I can't agree with you in case of user secrets, because that is used for development purpose. Anybody tried user secrets in production applications? Please advise

Sarathlal Saseendran
113 19.9k 5.3m
May 05, 2020
0

In this type of scenario we can use certificate, in which certificate deployed on production server and fingerprints of certification can be used in app settings

Ano Mepani
286 8.1k 1.7m
May 05, 2020
0

What is purpose of using azure key vault? What is best way to secure app using key vault?

Ano Mepani
286 8.1k 1.7m
Apr 27, 2020
0 1 Reply

Hi Ano, the main purpose of Key Vault is to store our sensitive information in Azure key vault using some service principal, client secret which fetch the value from secret. This will give some high level security to application but the problem is keeping the principal id and client secret. Better way to keep that values inside the database as per my experience.

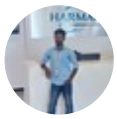
Sarathlal Saseendran
113 19.9k 5.3m
May 05, 2020
0

Hi Sarathlal , Nice Article but i have a doubt in this since i am new to azure platform. So my doub

Avi Jain
2031 40 0
Apr 08, 2020
1 2 Reply

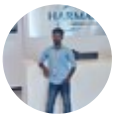
Hi Avi Jain, very good question. You can't use Key Vault in Web API application without any principal. I chose App Registration as the easy way. We will add the permissions like GET, LIST, DELETE to this app registration. Later in Web API we must use this app registration client id along with client secret which we already created in app registration. Hope you understood clearly. Please message me privately, if you need more clarifications.

Sarathlal Saseendran
113 19.9k 5.3m
Apr 12, 2020
0



Avi, I think Key vault cannot work individually. Here Sarathlal applied key vault with Simple web application. Am i right Sarathlal

Periyasamy Vellingirisamy
788 2.1k 217.2k
Apr 13, 2020
0



Nice one. Thanks for sharing

Periyasamy Vellingirisamy
788 2.1k 217.2k
Apr 08, 2020
1 1 Reply

Thank you very much Periyasamy Vellingiri

Sarathlal Saseendran
113 19.9k 5.3m
Apr 12, 2020
0

Very helpful article. :)

Jaimin Shethiya
605 3.3k 192.9k
Apr 06, 2020
1 1 Reply

Thank you Jaimin Shethiya for your feedback

Sarathlal Saseendran
113 19.9k 5.3m
Apr 12, 2020
1

Nice article on Asp.net API with Azure.

Dipa Mehta
183 12k 277.8k
Apr 06, 2020
2 1 Reply

Thank you very much Dipa Mehta

Sarathlal Saseendran
113 19.9k 5.3m
Apr 06, 2020
1

FEATURED ARTICLES

How To Upgrade to Windows 11

Exploring Subject <T> In Reactive Extensions For .Net

Micro Frontends With Webpack

What's New In iPhone 13

Understanding Synchronization Context Task.ConfigureAwait In Action

View All

TRENDING UP

01 Getting Started With ASP.NET Web API .NET 5

02 Everything That Every .NET Developer Needs To Know About Disposable Types - Properly Disposing Objects

03 How To Upgrade ASP.NET Core 2.1 To ASP.NET Core 3.1 version

04 What Happens Behind The Scenes When You Trade A Bitcoin?

05 Developing A Web App Using Angular 12, ASP.NET Core Web API And SQL Server

06 Build A ToDo List Application Using Angular, .Net 5.0 Web API And Microsoft SQL Server

07 Need For Site Reliability Engineering

08 Micro Frontends With Webpack

09 Two Ways To Create Minimal APIs In .NET 6

10 Generate Image Thumbnail using Azure Computer Vision and Function App

View All