**Use Azure Key Vault with a virtual machine in .NET**

The document shows you how to:

* Create a resource group.
* Create a key vault.
* Add a secret to the key vault.
* Retrieve a secret from the key vault.
* Create an Azure virtual machine.
* Enable a [managed identity](https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/overview) for the Virtual Machine.
* Assign permissions to the VM identity.

**Create a resource group and key vault**

az group create --name "myResourceGroup" -l "EastUS"

az keyvault create --name "<your-unique-keyvault-name>" -g "myResourceGroup"

**Populate your key vault with a secret**

To add a secret to your newly created key vault, use the following command:

az keyvault secret set --vault-name "<your-unique-keyvault-name>" --name "mySecret" --value "Success!"

**Create a Windows virtual machine with the Azure CLI**

az vm create \

--resource-group myResourceGroup \

--name myVM \

--image Win2019Datacenter \

--public-ip-sku Standard \

--admin-username azureuser

--admin-password Welcome@123456#

**Open port 80 for web traffic**

az vm open-port --port 80 --resource-group myResourceGroup --name myVM

Connect to the Windows VM and run the below command to install web server in windows VM

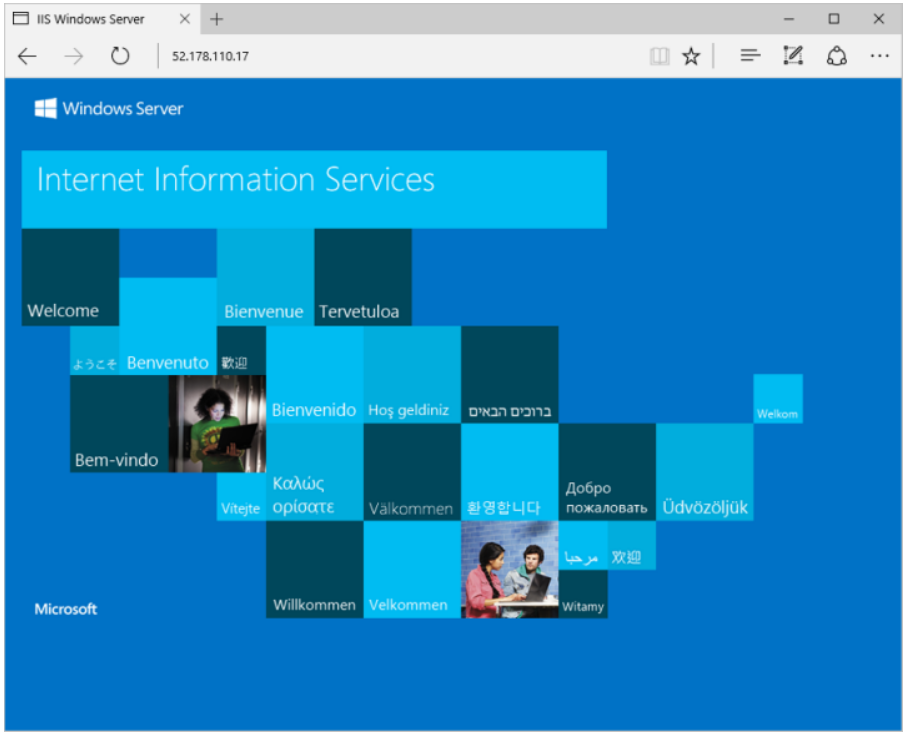
**Install web server**

To see your VM in action, install the IIS web server. Open a PowerShell prompt on the VM and run the following command:

Install-WindowsFeature -name Web-Server -IncludeManagementTools

**View the web server in action**

With IIS installed and port 80 now open on your VM from the Internet, use a web browser of your choice to view the default IIS welcome page. Use the public IP address of your VM obtained in a previous step. The following example shows the default IIS web site:



**Assign an identity to the VM**

Create a system-assigned identity for the virtual machine with the following example:

az vm identity assign --name <NameOfYourVirtualMachine> --resource-group <YourResourceGroupName>

**Assign permissions to the VM identity**

Assign the previously created identity permissions to your key vault with the [az keyvault set-policy](https://docs.microsoft.com/en-us/cli/azure/keyvault" \l "az-keyvault-set-policy) command:

az keyvault set-policy --name '<your-unique-key-vault-name>' --object-id <VMSystemAssignedIdentity> --secret-permissions get list set delete

**Connect to the VM and**

**Set up the console app**

Create a console app and install the required packages using the dotnet command.

**Install .NET Core**

To install .NET Core, go to the [.NET downloads](https://dotnet.microsoft.com/download) page.

**Create and run a sample .NET app**

Open a command prompt.

You can print "Hello World" to the console by running the following commands:

dotnet new console -n keyvault-console-app

cd keyvault-console-app

dotnet run

**Install the package**

From the console window, install the Azure Key Vault Secrets client library for .NET:

dotnet add package Azure.Security.KeyVault.Secrets

For this quickstart, you will need to install the following identity package to authenticate to Azure Key Vault:

dotnet add package Azure.Identity

**Edit the console app**

Open the *Program.cs* file and add these packages:

using System;

using Azure.Core;

using Azure.Identity;

using Azure.Security.KeyVault.Secrets;

Add these lines, updating the URI to reflect the vaultUri of your key vault. Below code is using ['DefaultAzureCredential()'](https://docs.microsoft.com/en-us/dotnet/api/azure.identity.defaultazurecredential) for authentication to key vault, which is using token from application managed identity to authenticate. It is also using exponential backoff for retries in case of key vault is being throttled.

class Program

{

static void Main(string[] args)

{

string secretName = "mySecret";

string keyVaultName = "<your-key-vault-name>";

var kvUri = "https://<your-key-vault-name>.vault.azure.net";

SecretClientOptions options = new SecretClientOptions()

{

Retry =

{

Delay= TimeSpan.FromSeconds(2),

MaxDelay = TimeSpan.FromSeconds(16),

MaxRetries = 5,

Mode = RetryMode.Exponential

}

};

var client = new SecretClient(new Uri(kvUri), new DefaultAzureCredential(),options);

Console.Write("Input the value of your secret > ");

string secretValue = Console.ReadLine();

Console.Write("Creating a secret in " + keyVaultName + " called '" + secretName + "' with the value '" + secretValue + "` ...");

client.SetSecret(secretName, secretValue);

Console.WriteLine(" done.");

Console.WriteLine("Forgetting your secret.");

secretValue = "";

Console.WriteLine("Your secret is '" + secretValue + "'.");

Console.WriteLine("Retrieving your secret from " + keyVaultName + ".");

KeyVaultSecret secret = client.GetSecret(secretName);

Console.WriteLine("Your secret is '" + secret.Value + "'.");

System.Threading.Thread.Sleep(5000);

Console.WriteLine(" done.");

}

}

Then run below command

dotnet run