

GlobalSign & Microsoft Azure Key Vault Integration Guide

Secure Key Management for cloud apps

GLOBALSIGN INTEGRATION GUIDE



INTRODUCTION

This document will walk you through the process of issuing a SSL certificate from your GlobalSign Managed SSL account into an Azure Key Vault. If you do not currently have a GlobalSign account, please [contact us here](#).

Requirements

You will need to have Azure PowerShell Version 2.2.0 or greater which is available for download here.

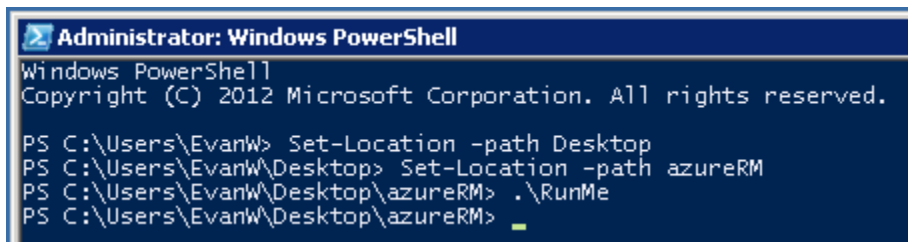
<https://github.com/Azure/azure-powershell/releases>

Steps

Run GlobalSign Key Vault folder via PowerShell

Unzip attached folder (GlobalSign.KeyVault) and run the RunMe.ps1 from the attached folder

(1-RunMe_PowerShell_1)

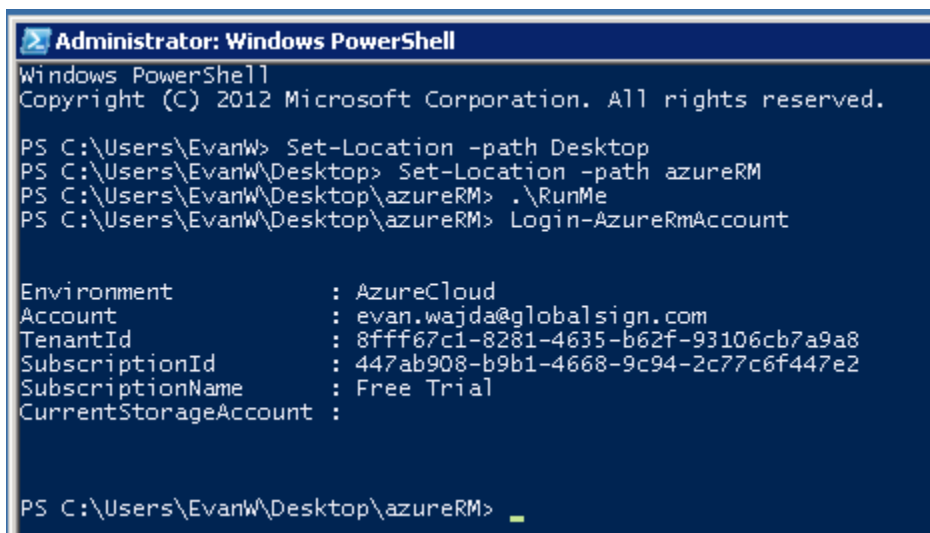


```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Users\EvanW> Set-Location -path Desktop
PS C:\Users\EvanW\Desktop> Set-Location -path azureRM
PS C:\Users\EvanW\Desktop\azureRM> .\RunMe
PS C:\Users\EvanW\Desktop\azureRM> _
```

Login-AzureRmAccount

(2-RunMe & Azure_Log_In_PowerShell)



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

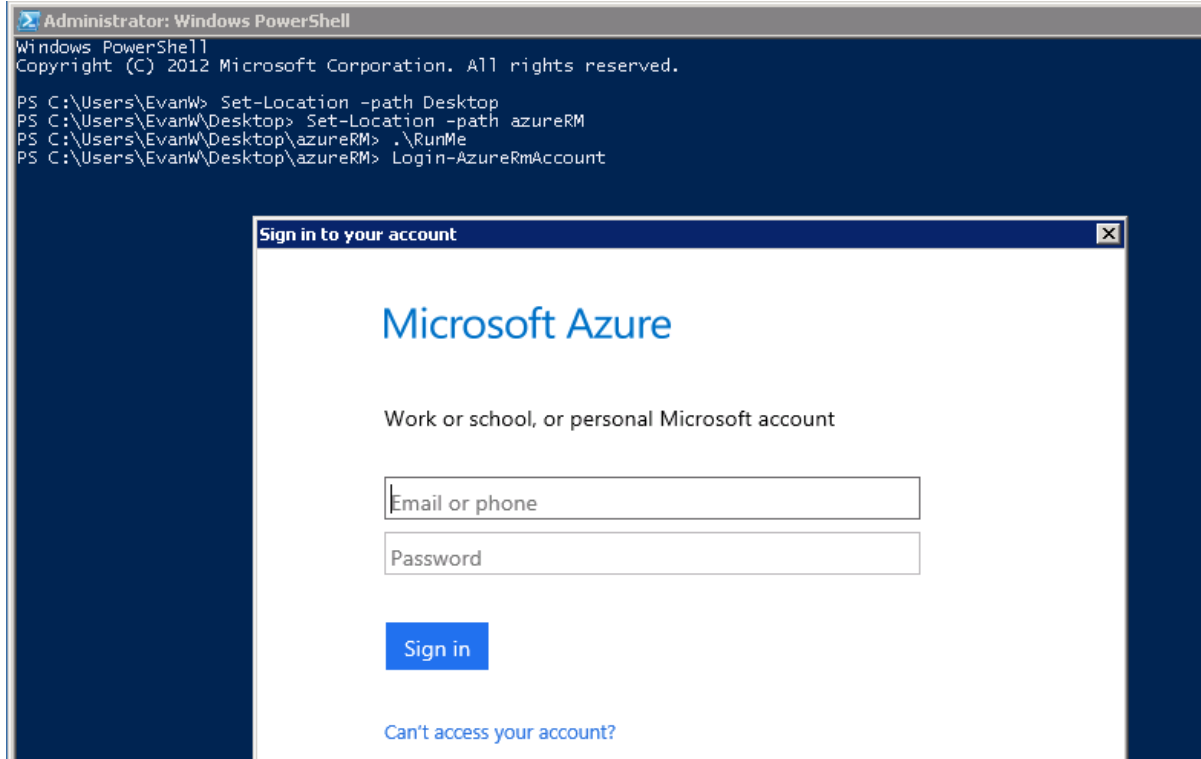
PS C:\Users\EvanW> Set-Location -path Desktop
PS C:\Users\EvanW\Desktop> Set-Location -path azureRM
PS C:\Users\EvanW\Desktop\azureRM> .\RunMe
PS C:\Users\EvanW\Desktop\azureRM> Login-AzureRmAccount

Environment      : AzureCloud
Account          : evan.wajda@globalsign.com
TenantId         : 8fff67c1-8281-4635-b62f-93106cb7a9a8
SubscriptionId   : 447ab908-b9b1-4668-9c94-2c77c6f447e2
SubscriptionName : Free Trial
CurrentStorageAccount :

PS C:\Users\EvanW\Desktop\azureRM> _
```

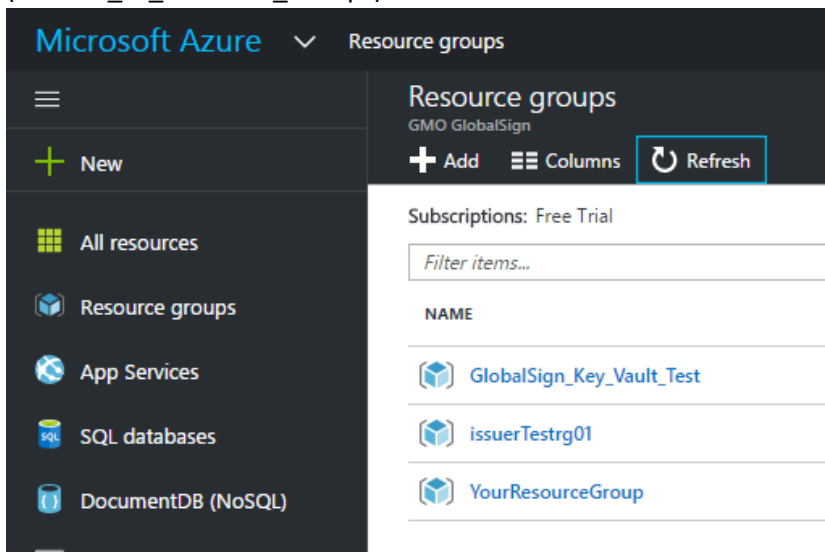
You will then be prompted for your Microsoft Azure Account credentials. If you don't have an Azure account you will need to create one here <https://azure.microsoft.com/en-us/>

(3-Azure_Log_In_PoweShell_2)



You will now need to set reference group. If you have a resource group already created you would like to use, you want to reference that now or create a new reference group. You can access and add additional reference groups in the Azure portal.

(4-Azure_UI_Resource_Groups)



Or in PowerShell.

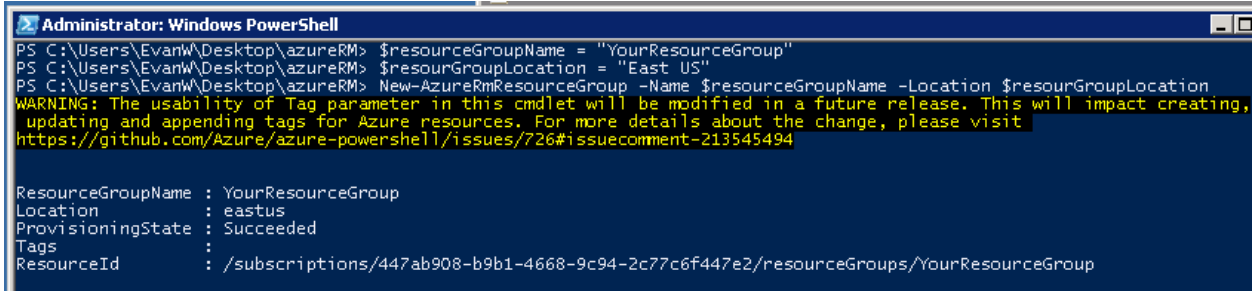
```
$resourceGroupName = "YourResourceGroup"
```

```
$resourceGroupLocation = "Your Location"
```

```
New-AzureRmResourceGroup -Name $resourceGroupName -Location $resourceGroupLocation
```

Note: A list of locations are available in your Azure Portal.

(5-Create_Resource_Group_PowerShell)

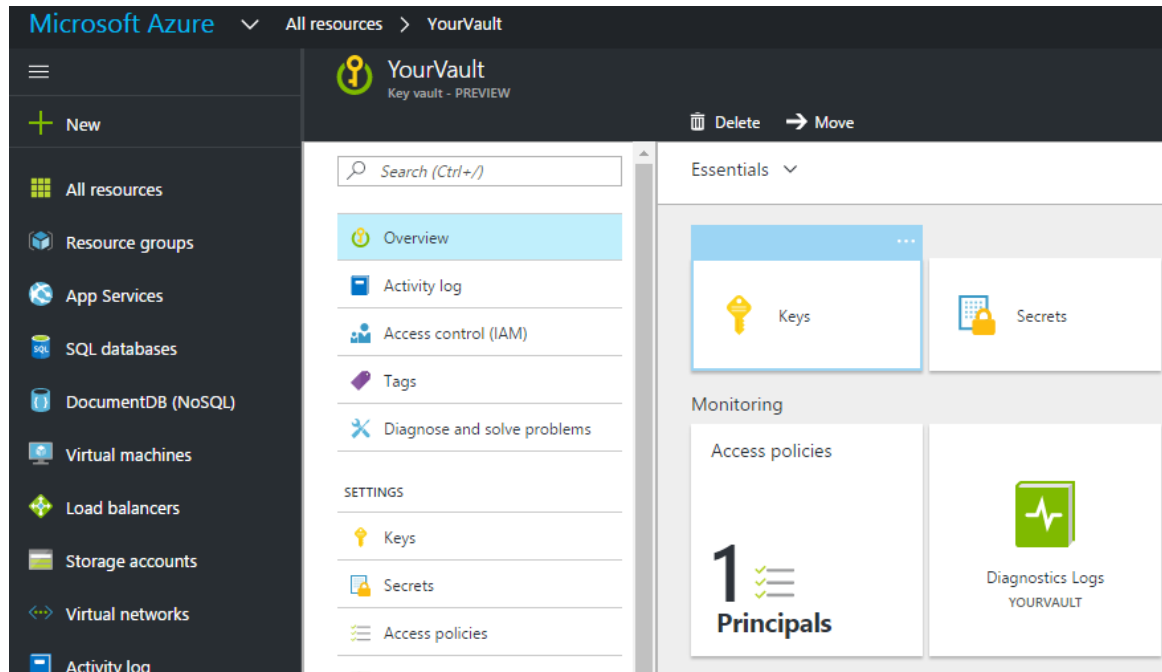


```
Administrator: Windows PowerShell
PS C:\Users\EvanW\Desktop\azureRM> $resourceGroupName = "YourResourceGroup"
PS C:\Users\EvanW\Desktop\azureRM> $resourceGroupLocation = "East US"
PS C:\Users\EvanW\Desktop\azureRM> New-AzureRmResourceGroup -Name $resourceGroupName -Location $resourceGroupLocation
WARNING: The usability of Tag parameter in this cmdlet will be modified in a future release. This will impact creating,
updating and appending tags for Azure resources. For more details about the change, please visit
https://github.com/Azure/azure-powershell/issues/726#issuecomment-213545494

ResourceGroupName : YourResourceGroup
Location           : eastus
ProvisioningState  : Succeeded
Tags               :
ResourceId         : /subscriptions/447ab908-b9b1-4668-9c94-2c77c6f447e2/resourceGroups/YourResourceGroup
```

If you have a vault already created you would like to use, you want to reference that now or create a new vault. You can access and add additional vaults in the Azure portal.

(6-Create_Vault_Azure_UI)



Or in PowerShell.

\$vaultName = "YourVaultNameHere"

\$vaultLocation = "Your Location"

New-AzureRmKeyVault -VaultName \$vaultName -ResourceGroupName \$resourceGroupName - Location \$vaultLocation -Sku Premium

Note: A list of locations are available in your Azure Portal.

(7-Create_Vault_PS)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $vaultName = "YourVault"
PS C:\Users\Evan\Desktop\azureRM> $vaultLocation = "East US"
PS C:\Users\Evan\Desktop\azureRM> New-AzureRmKeyVault -VaultName $vaultName -ResourceGroupName $resourceGroupName -Location $vaultLocation -Sku Premium
WARNING: The usage of Tag parameter in this cmdlet will be modified in a future release. This will impact creating, updating and appending tags for Azure resources. For more details about the change, please visit https://github.com/Azure/azure-powershell/issues/726#issuecomment-213545494

Vault Name                               : YourVault
Resource Group Name                      : YourResourceGroup
Location                                 : East US
Resource ID                               : /subscriptions/447ab908-b9b1-4668-9c94-2c77c6f447e2/resourceGroups/YourResourceGroup/providers/Microsoft.KeyVault/vaults/YourVault
Vault URI                                 : https://YourVault.vault.azure.net
Tenant ID                                 : 8fff67c1-8281-4635-b62f-93106cb7a9a8
SKU                                       : Premium
Enabled For Deployment?                  : False
Enabled For Template Deployment?         : False
Enabled For Disk Encryption?             : False
Access Policies                          :
                                           Tenant ID           : 8fff67c1-8281-4635-b62f-93106cb7a9a8
                                           Object ID           : b099492e-6bfb-48f4-80c5-25491fe29c96
                                           Application ID      :
                                           Display Name       : Evan Wajda (evan.wajda@globalsign.com)
                                           Permissions to Keys : get, create, delete, list, update, import, backup, restore
                                           Permissions to Secrets : all
                                           Permissions to Certificates : all
  
```

You will now want to create an Administrator for the issuer. This will assign a point of contact for the issued certificate.

\$firstName = "KeyVault"

\$lastName = "Issuer"

\$phoneNumber = "8008675309"

\$emailAddress = "youremail@domain.com"

\$admin = New-AzureKeyVaultCertificateAdministratorDetails -FirstName \$firstName -LastName \$lastName -PhoneNumber \$phoneNumber -EmailAddress \$emailAddress

(8-Admin_User)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $firstName = "KeyVault"
PS C:\Users\Evan\Desktop\azureRM> $lastName = "Issuer"
PS C:\Users\Evan\Desktop\azureRM> $phoneNumber = "8008675309"
PS C:\Users\Evan\Desktop\azureRM> $emailAddress = "email@domain.com"
PS C:\Users\Evan\Desktop\azureRM> New-AzureKeyVaultCertificateAdministratorDetails -FirstName $firstName -LastName $lastName -PhoneNumber $phoneNumber -EmailAddress $emailAddress

FirstName                LastName                EmailAddress            PhoneNumber
-----
KeyVault                 Issuer                 email@domain.com       8008675309

PS C:\Users\Evan\Desktop\azureRM> $admin = New-AzureKeyVaultCertificateAdministratorDetails -FirstName $firstName -LastName $lastName -PhoneNumber $phoneNumber -EmailAddress $emailAddress
PS C:\Users\Evan\Desktop\azureRM>
  
```

You will now want to create an Organization for the issuer.

\$org = New-AzureKeyVaultCertificateOrganizationDetails -AdministratorDetails \$admin

(9-Create_Organization)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $org = New-AzureKeyVaultCertificateOrganizationDetails -AdministratorDetails $admin
PS C:\Users\Evan\Desktop\azureRM>
  
```

Next, you will want to create an issuer. The “apiKey” is your GlobalSign password and the “accountId” is your GlobalSign User ID assigned when creating your GlobalSign account.

\$apiKey = "YourGlobalSignUserPassword"

\$secureApiKey = ConvertTo-SecureString \$apiKey -AsPlainText -Force

\$accountId = "PAR123456_YourUserID"

\$issuerName = "YourIssuerName"

Add-AzureKeyVaultCertificateIssuer -VaultName \$vaultName -IssuerName \$issuerName -IssuerProvider GlobalSign -AccountId \$accountId -ApiKey \$secureApiKey -OrganizationDetails \$org

(10-Create_Issuer)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $apiKey = "YourGlobalSignPassword"
PS C:\Users\Evan\Desktop\azureRM> $secureApiKey = ConvertTo-SecureString $apiKey -AsPlainText -Force
PS C:\Users\Evan\Desktop\azureRM> $accountId = "PAR123456_YourUserID"
PS C:\Users\Evan\Desktop\azureRM> $issuerName = "YourIssuerName"
PS C:\Users\Evan\Desktop\azureRM> Add-AzureKeyVaultCertificateIssuer -VaultName $vaultName -IssuerName $issuerName -IssuerProvider GlobalSign -AccountId $accountId -ApiKey $secureApiKey -OrganizationDetails $org
PS C:\Users\Evan\Desktop\azureRM>
  
```

Then you will want to create a certificate policy. This is where you set the common name (from your pre-validated GlobalSign domains) to be issued. You will also specify the validity period (in months) and how many days before expiry you would like the certificate to be renewed.

\$policy = New-AzureKeyVaultCertificatePolicy -SecretContentType application/x-pkcs12 -SubjectName "CN=healthytest.xyz" -ValidityInMonths 12 -IssuerName \$issuerName -RenewAtNumberOfDaysBeforeExpiry 60

CN=’s Common Name (FQDN) of certificate

(11-Create_Policy)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $policy = New-AzureKeyVaultCertificatePolicy -SecretContentType application/x-pkcs12 -SubjectName "CN=yourdomain.com" -ValidityInMonths 12 -IssuerName $issuerName -RenewAtNumberOfDaysBeforeExpiry 60
PS C:\Users\Evan\Desktop\azureRM>
  
```

Next you will request for certificate enrollment. If successful, you will get a “Status” of inProgress as well as a confirmation under “StatusDetails” that the Pending certificate is being created and the request is in process as well as the CSR.

\$certificateName = "globalsignTrial01"

Add-AzureKeyVaultCertificate -VaultName \$vaultName -CertificateName \$certificateName -CertificatePolicy \$policy

Note: If you receive an Error, please send the Error Message along with your account information to support@globalsign.com. Please reference “Azure Key Vault Error” in the subject.

(12-Request-Certificate)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> $certificateName = "YourCertificateName"
PS C:\Users\Evan\Desktop\azureRM> Add-AzureKeyVaultCertificate -VaultName $vaultName -CertificateName $certificateName
-CertificatePolicy $policy

Id                : https://yourvault.vault.azure.net/certificates/YourCertificateName/pending
Status            : inProgress
StatusDetails    : Pending certificate created. Certificate request is in progress. This may take some time
                  based on the issuer provider. Please check again later.
RequestId        : e03a1291757b477a814bcb356a5ba615
Target           :
Issuer           : globalsign1
CancellationRequested : False
CertificateSigningRequest : MIICQjCCAZICAAwGjEYMBYGA1UEAxMPYXd1c29tZTlWMDguY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgk
CAQEAFANR1nPJJGpcQB49kU4DYRG1Wa8CW9Rb00oFkmpH48ZRW/Yqb+v1YxTLMZwutW+nS+qDxw5FU1fA6upXLB60
bqI812PNuTZwo1z7pCM0b4CHXTq479gRqd8jyAgOpWkTK/xb0t4Wkn1n1ZWhgSYJPSIKjCjULaxwWIW49PV51M8YVPj
puWdym1UjKyikkPhMOctHKWjZ79woKPCQB84aAo9hkPXUblZYFvJge6+aPdjqeWtGu4UG9ajCUSrIc15CbXBwRHXZ
sCH8eQFhezFexEmxZST3jLkABEbHmQWjg0dyxRLOBnDKCAiMtgsxcGrY21L5Y778Pafk0gM9zQIDAQABoEswSQYjKoZ
IhvcNAQK0MTwwQjA0BgNVHQ8BAF8EBAMCBAAwHwYDVR01BBYwFAyIKwYBBQUHAWEGCCsGAUQUFwMCMAMGA1UdEwQCA
AwDQYjKoZIHvcNAQELBQADggEBALgbF1WBkTpcGef7/wf6NjtX91NquzEzZLur7w2iBcs+tnP+sYzhSM8S+Tedgb/q
XMs91uG5A1MwRh2ejdKSDDly951ft2PdLJ1a0hI7wQkm/BQkqAI4WjMUTHdc+xVT1b96xPsvW+PT6zk7kLA6zptn8d
ZJzAYAV22YqTjmSxtW9XFg2cw1EjwXTvxw1cSAFJXRvkDQePdIYfWrQoGc8aIeuSzM9TSzb1jGc9G+zmuR1aVpYbJGA
wTixSZJS/6+KsZwvK7QNZ13sYfy0uWHnPw80S3NMIVax/w+OCmkf1bjAzfH0wd14JYpNuetfAJQvfZJMmBqVBU8oLk
6QIWA=

ErrorCode        :
ErrorMessage     :
  
```

Now you will want to confirm the certificate has been issued

Get-AzureKeyVaultCertificate -VaultName \$vaultName -CertificateName \$certificateName

(13-Certificate_Creation_Confirmation)

```

Administrator: Windows PowerShell
PS C:\Users\Evan\Desktop\azureRM> Get-AzureKeyVaultCertificate -VaultName $vaultName -CertificateName $certificateName

Name                : YourCertificateName
Certificate          : [Subject]
                   : CN=awesome2008.com, O="GlobalSign, Inc (For test purposes only)", L=Portsmouth, S=NH, C=US
                   :
                   : [Issuer]
                   : CN=GlobalSign Organization Validation CA - SHA256 - G2, O=GlobalSign nv-sa, C=BE
                   :
                   : [Serial Number]
                   : 3989F2A7006187F448135DC8
                   :
                   : [Not Before]
                   : 9/23/2016 12:04:38 AM
                   :
                   : [Not After]
                   : 9/24/2017 12:04:38 AM
                   :
                   : [Thumbprint]
                   : 5F56CE7531B11A1EFF09230317CBE540C3B11870

Id                : https://yourvault.vault.azure.net:443/certificates/YourCertificateName/4f40f744ee484804b5e0e08804520f38
KeyId             : https://yourvault.vault.azure.net:443/keys/YourCertificateName/4f40f744ee484804b5e0e08804520f38
SecretId          : https://yourvault.vault.azure.net:443/secrets/YourCertificateName/4f40f744ee484804b5e0e08804520f38
Thumbprint       : 5F56CE7531B11A1EFF09230317CBE540C3B11870
Tags              :
Enabled           : True
Created           : 9/23/2016 12:04:42 AM
Updated           : 9/23/2016 12:04:42 AM
  
```

Note: If the certificate is still pending issuance you can check the status enrollment by using the script below.

Get-AzureKeyVaultCertificateOperation -VaultName \$vaultName -CertificateName \$certificateName

(14-Certificate_Status)

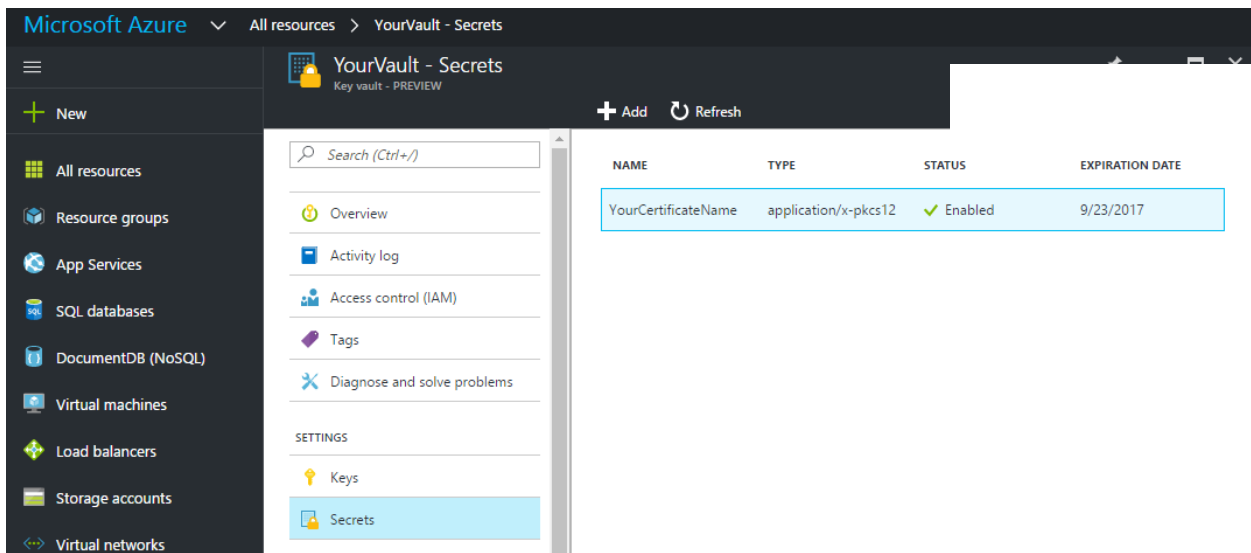
```

Administrator: Windows PowerShell
PS C:\Users\EvanW\Desktop\azureRM> Get-AzureKeyVaultCertificateOperation -VaultName $vaultName -CertificateName $certifi
cateName

Id                : https://yourvault.vault.azure.net/certificates/YourCertificateName/pending
Status            : completed
StatusDetails    :
RequestId        : e03a1291757b477a814bcb356a5ba615
Target           : https://yourvault.vault.azure.net/certificates/YourCertificateName
Issuer           : globalSign1
CancellationRequested : False
CertificateSigningRequest : MIICqjCCAZICAQAwGjEYMBYGA1UEAxMPYXd1c29tZTlwMDguY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCQA8AMIIBCgK
CAQEAxFEANR1nPJJGpcQB49kU4DYRG1wa8CW9Rb00oFkmpH48ZRW/Yqb+ViYxTLM2wutW+nS+gDXw5FU1fA6upXLB60
bqI8l2PNuIT2woiz7pCM0b4CHXTq479gRad8jyAg0plwkTK/xb0t4Wkn1n1ZMhgSYJPSIKjCjULaxwWIW49PV51M8YVPj
puWdym1UjKy1khPhM0CtHKWjZ79woKPCQBP84aAo9hkPXUblZYFvJge6+aPdjQeWtGu4UG9ajCUSrIc15CbxBwRHVXZ
sCH8eQFhezFexEmxZST3jLkABEbHmQWjg0dyxRL0BnDKCA1MtgsxcGrY21L5Y778Pafk0gM9zQIDAQABoEswSQYKoz
IhvcNAQKODMtwOjA0BgNVHQ8BAF8EBAMCBaAwHQYDVRO1BBYwFAyIKwYBBQUHAWEGCCsGAQUFwBwMCMakGA1UdEwQCMA
AwDQYJKoZIhvcNAQELBQADggEBALgbF1WBkTpcCGeF7/wf6NjTX9lNquzEzZLur7w2iBcs+tnP+sYzhSM8S+TEdgb/q
XMsm91uG5A1MwRh2ejdKSDd1y951ft2PdLJ1a0hI7w0km/BQkqAI4WjMUTHdc+XVT1b96XPsvW+PT6zk7kLA6zptn8d
ZJzAYAV22YqTjmSxtW9XFg2cw1E1WXTvxwLcSAFJXrvkDQePdIYfWrQoGc8aIeuszM9TSzb1jGc9G+zmuR1aVpYbJGA
wT1xSZJS/6+KsZwwK7QN213sYfy0uWHnPw80S3NMIvAx/w+OCmkf1bjAzfH0wd14JYpNuetfAJQvFZJMmBqVBU8oLk
6QIWA=
  
```

Certificate issued in Azure Portal

(15-Certificate_Azure_UI)



ABOUT GLOBALSIGN

GlobalSign is a leading provider of trusted identity and security solutions enabling businesses, large enterprises, cloud service providers and IoT innovators around the world to secure online communications, manage millions of verified digital identities and automate authentication and encryption. Its high-scale PKI and Identity and Access Management (IAM) solutions support the billions of services, devices, people and things comprising the Internet of Everything (IoE).



US: +1 877 775 4562
UK: +144 1622 766766
EU: +32 16 89 19 00
sales@globalsign.com

For additional information,
please visit www.globalsign.com