
## *Windows PowerShell Get-Help on Cmdlet 'Set-AzKeyVaultAccessPolicy'*

***PS:\>Get-HELP Set-AzKeyVaultAccessPolicy -Full***

NAME

Set-AzKeyVaultAccessPolicy

SYNOPSIS

Grants or modifies existing permissions for a user, application, or security group to perform operations with a key vault.

SYNTAX

Set-AzKeyVaultAccessPolicy [-VaultName] <System.String> [[-ResourceGroupName] <System.String>] [-ApplicationId <System.Nullable`1[System.Guid]>]

[-BypassObjectIdValidation] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -ObjectId <System.String>

[-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage

<System.String[]>] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem> [-ApplicationId <System.Nullable`1[System.Guid]>]

[-BypassObjectIdValidation] [-DefaultProfile

<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -ObjectId <System.String>
[-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage
<System.String[]>] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-ResourceId] <System.String> [-ApplicationId <System.Nullable`1[System.Guid]>] [-BypassObjectIdValidation] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -ObjectId <System.String> [-PassThru] [-PermissionsToCertificates
<System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId
<System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-VaultName] <System.String> [[-ResourceGroupName] <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -EmailAddress <System.String> [-PassThru] [-PermissionsToCertificates
<System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId
<System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem> [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] -EmailAddress <System.String> [-PassThru] [-PermissionsToCertificates
<System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId
<System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-ResourceId] <System.String> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]
-EmailAddress <System.String> [-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets

<System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-VaultName] <System.String> [[-ResourceGroupName] <System.String>] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-EnabledForDeployment] [-EnabledForDiskEncryption] [-EnabledForTemplateDeployment] [-PassThru] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-EnabledForDeployment] [-EnabledForDiskEncryption] [-EnabledForTemplateDeployment] [-PassThru] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-ResourceId] <System.String> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-EnabledForDeployment] [-EnabledForDiskEncryption] [-EnabledForTemplateDeployment] [-PassThru] [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] -ServicePrincipalName <System.String> [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-PassThru]

[-PermissionsToCertificates <System.String[]>]
[-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId <System.String>]
-UserPrincipalName <System.String> [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-VaultName] <System.String> [[-ResourceGroupName] <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-PassThru] [-PermissionsToCertificates <System.String[]>]
[-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] [-SubscriptionId <System.String>]
-UserPrincipalName <System.String> [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-VaultName] <System.String> [[-ResourceGroupName] <System.String>] [-DefaultProfile
<Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-PassThru] [-PermissionsToCertificates <System.String[]>]
[-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage <System.String[]>] -ServicePrincipalName <System.String>
[-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-ResourceId] <System.String> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]
[-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage
<System.String[]>] -ServicePrincipalName <System.String> [-SubscriptionId <System.String>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultAccessPolicy [-ResourceId] <System.String> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>]
[-PassThru] [-PermissionsToCertificates <System.String[]>] [-PermissionsToKeys <System.String[]>] [-PermissionsToSecrets <System.String[]>] [-PermissionsToStorage
<System.String[]>] [-SubscriptionId <System.String>] -UserPrincipalName <System.String> [-Confirm] [-WhatIf] [<CommonParameters>]

DESCRIPTION

The Set-AzKeyVaultAccessPolicy cmdlet grants or modifies existing permissions for a user, application, or security group to perform the specified operations with a

key vault. It does not modify the permissions that other users, applications, or security groups have on the key vault. If you are setting permissions for a security

group, this operation affects only users in that security group. The following directories must all be the same Azure directory: - The default directory of the Azure

subscription in which the key vault resides.

- The Azure directory that contains the user or application group that you are granting permissions to.

Examples of scenarios when these conditions are not met and this cmdlet will not work are: - Authorizing a user from a different organization to manage your key

vault. Each organization has its own directory. - Your Azure account has multiple directories. If you register an application in a directory other than the default

directory, you cannot authorize that application to use your key vault. The application must be in the default directory. Note that although specifying the resource

group is optional for this cmdlet, you should do so for better performance.

The cmdlet may call below Microsoft Graph API according to input parameters:

- GET /directoryObjects/{id}

- GET /users/{id}

- GET /users

- GET /servicePrincipals/{id}

- GET /servicePrincipals

- GET /groups/{id}

> [!NOTE] > When using a service principal to grant access policy permissions, you must use the `-BypassObjectIdValidation` parameter.

PARAMETERS

-ApplicationId <System.Nullable`1[System.Guid]>

For future use.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-BypassObjectIdValidation <System.Management.Automation.SwitchParameter>

Enables you to specify an object ID without validating that the object exists in Microsoft Entra ID. Use this parameter only if you want to grant access to your

key vault to an object ID that refers to a delegated security group from another Azure tenant.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>

The credentials, account, tenant, and subscription used for communication with azure

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EmailAddress <System.String>

Specifies the user email address of the user to whom to grant permissions. This email address must exist in the directory associated with the current subscription

and be unique.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-EnabledForDeployment <System.Management.Automation.SwitchParameter>

Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when

creating a virtual machine.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-EnabledForDiskEncryption <System.Management.Automation.SwitchParameter>

Enables the Azure disk encryption service to get secrets and unwrap keys from this key vault.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-EnabledForTemplateDeployment <System.Management.Automation.SwitchParameter>

Enables Azure Resource Manager to get secrets from this key vault when this key vault is referenced in a template deployment.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-InputObject <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem>

Key Vault Object

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-ObjectId <System.String>

Specifies the object ID of the user or service principal in Microsoft Entra ID for which to grant permissions. Its value is in the format of GUID.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PassThru <System.Management.Automation.SwitchParameter>

Returns an object representing the item with which you are working. By default, this cmdlet does not generate any output.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-PermissionsToCertificates <System.String[]>

Specifies an array of certificate permissions to grant to a user or service principal. 'All' will grant all the permissions except 'Purge' The acceptable values

for this parameter: - All

- Get

- List

- Delete

- Create

- Import

- Update

- Managecontacts

- Getissuers

- Listissuers

- Setissuers

- Deleteissuers

- Manageissuers

- Recover

- Backup

- Restore

- Purge

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PermissionsToKeys <System.String[]>

Specifies an array of key operation permissions to grant to a user or service principal. 'All' will grant all the permissions except 'Purge' The acceptable values

for this parameter: - All

- Decrypt

- Encrypt

- UnwrapKey

- WrapKey

- Verify

- Sign

- Get

- List

- Update

- Create

- Import

- Delete

- Backup

- Restore

- Recover

- Purge

- Rotate

Required? false
Position? named
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-PermissionsToSecrets <System.String[]>

Specifies an array of secret operation permissions to grant to a user or service principal. 'All' will grant all the permissions except 'Purge' The acceptable

values for this parameter: - All

- Get

- List

- Set

- Delete

- Backup

- Restore

- Recover

- Purge

Required? false
Position? named
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-PermissionsToStorage <System.String[]>

Specifies managed storage account and SaS-definition operation permissions to grant to a user or service principal. 'All' will grant all the permissions except

'Purge' The acceptable values for this parameter: - all

- get

- list

- delete

- set

- update

- regeneratekey

- getsas

- listsas

- deletesas

- setsas

- recover

- backup

- restore

- purge

Required? false
Position? named
Default value None
Accept pipeline input? False
Accept wildcard characters? false

-ResourceGroupName <System.String>

Specifies the name of a resource group.

Required? false

Position? 1

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-ResourceId <System.String>

Key Vault Resource Id

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-ServicePrincipalName <System.String>

Specifies the service principal name of the application to which to grant permissions. Specify the application ID, also known as client ID, registered for the

application in Microsoft Entra ID. The application with the service principal name that this parameter specifies must be registered in the Azure directory that

contains your current subscription.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SubscriptionId <System.String>

The ID of the subscription. By default, cmdlets are executed in the subscription that is set in the current context. If the user specifies another subscription,

the current cmdlet is executed in the subscription specified by the user. Overriding subscriptions only take effect during the lifecycle of the current cmdlet. It

does not change the subscription in the context, and does not affect subsequent cmdlets.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-UserPrincipalName <System.String>

Specifies the user principal name of the user to whom to grant permissions. This user principal name must exist in the directory associated with the current

subscription.

Required? true

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-VaultName <System.String>

Specifies the name of a key vault. This cmdlet modifies the access policy for the key vault that this parameter specifies.

Required? true

Position? 0

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug,

ErrorAction, ErrorVariable, WarningAction, WarningVariable,

OutBuffer, PipelineVariable, and OutVariable. For more information, see

about_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultIdentityItem

System.String

OUTPUTS

Microsoft.Azure.Commands.KeyVault.Models.PSKeyVault

NOTES

Example 1: Grant permissions to a user for a key vault and modify the permissions

```
Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -UserPrincipalName 'PattiFuller@contoso.com'
-PermissionsToKeys create,import,delete,list -PermissionsToSecrets
set,delete -PassThru

Vault Name                      : Contoso03Vault
Resource Group Name             : myrg
Location                        : westus
Resource ID                     : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myrg/providers
                                  /Microsoft.KeyVault/vaults/contoso03vault
Vault URI                       : https://contoso03vault.vault.azure.net/
Tenant ID                       : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
SKU                             : Standard
Enabled For Deployment?         : True
Enabled For Template Deployment? : False
Enabled For Disk Encryption?    : False
Soft Delete Enabled?            : True
Access Policies                 :
                                  Tenant ID                       : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                                  Object ID                       : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                                  Application ID                  :
                                  Display Name                    : User Name (username@microsoft.com)
                                  Permissions to Keys             : create, import, delete, list
                                  Permissions to Secrets          : set, delete
                                  Permissions to Certificates     :
                                  Permissions to (Key Vault Managed) Storage :

Tags                            :
```

```
Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -UserPrincipalName 'PattiFuller@contoso.com'
-PermissionsToSecrets set,delete,get -PassThru

   Vault Name                  : Contoso03Vault
   Resource Group Name         : myrg
   Location                    : westus
   Resource ID                 : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myrg/providers
                                 /Microsoft.KeyVault/vaults/contoso03vault
   Vault URI                   : https://contoso03vault.vault.azure.net/
   Tenant ID                   : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
   SKU                         : Standard
   Enabled For Deployment?     : True
   Enabled For Template Deployment? : False
   Enabled For Disk Encryption?  : False
   Soft Delete Enabled?        : True
   Access Policies             :
                               Tenant ID                  : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                               Object ID                  : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                               Application ID             :
                               Display Name               : User Name (username@microsoft.com)
                               Permissions to Keys        : create, import, delete, list
                               Permissions to Secrets     : set, delete, get
                               Permissions to Certificates :
                               Permissions to (Key Vault Managed) Storage :

   Tags                        :

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -UserPrincipalName 'PattiFuller@contoso.com'
-PermissionsToKeys @() -PassThru

   Vault Name                  : Contoso03Vault
   Resource Group Name         : myrg
   Location                    : westus
```

```
Resource ID                : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/myrg/providers
                           /Microsoft.KeyVault/vaults/contoso03vault
Vault URI                  : https://contoso03vault.vault.azure.net/
Tenant ID                  : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
SKU                        : Standard
Enabled For Deployment?    : True
Enabled For Template Deployment? : False
Enabled For Disk Encryption?     : False
Soft Delete Enabled?       : True
Access Policies            :
                           Tenant ID                  : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                           Object ID                  : xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
                           Application ID             :
                           Display Name               : User Name (username@microsoft.com)
                           Permissions to Keys        :
                           Permissions to Secrets     : set, delete, get
                           Permissions to Certificates :
                           Permissions to (Key Vault Managed) Storage :

Tags                       :
```

The first command grants permissions for a user in your Microsoft Entra ID, PattiFuller@contoso.com, to perform operations on keys and secrets with a key vault named

Contoso03Vault. The PassThru parameter results in the updated object being returned by the cmdlet. The second command modifies the permissions that were granted to

PattiFuller@contoso.com in the first command, to now allow getting secrets in addition to setting and deleting them. The permissions to key operations remain

unchanged after this command. The final command further modifies the existing permissions for PattiFuller@contoso.com to remove all permissions to key operations. The

permissions to secret operations remain unchanged after this command.

Example 2: Grant permissions for an application service principal to read and write secrets

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -ServicePrincipalName 'http://payroll.contoso.com' -PermissionsToSecrets Get,Set

This command grants permissions for an application for a key vault named Contoso03Vault. The ServicePrincipalName parameter specifies the application. The application

must be registered in your Microsoft Entra ID. The value of the ServicePrincipalName parameter must be either the service principal name of the application or the

application ID GUID. This example specifies the service principal name `http://payroll.contoso.com`, and the command grants the application permissions to read and

write secrets.

Example 3: Grant permissions for an application using its object ID

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -ObjectId 34595082-9346-41b6-8d6b-295a2808b8db -PermissionsToSecrets Get,Set

This command grants the application permissions to read and write secrets. This example specifies the application using the object ID of the service principal of the

application.

---- Example 4: Grant permissions for a user principal name ----

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -UserPrincipalName 'PattiFuller@contoso.com' -PermissionsToSecrets Get,List,Set

This command grants get, list, and set permissions for the specified user principal name for access to secrets.

Example 5: Enable secrets to be retrieved from a key vault by the Microsoft.Compute resource provider

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso03Vault' -ResourceGroupName 'Group14' -EnabledForDeployment

This command grants the permissions for secrets to be retrieved from the Contoso03Vault key vault by the Microsoft.Compute resource provider.

------- Example 6: Grant permissions to a security group -------

Get-AzADGroup

Set-AzKeyVaultAccessPolicy -VaultName 'myownvault' -ObjectId (Get-AzADGroup -SearchString 'group2')[0].Id -PermissionsToKeys get, set -PermissionsToSecrets get, set

The first command uses the Get-AzADGroup cmdlet to get all Active Directory groups. From the output, you see 3 groups returned, named group1 , group2 , and group3 .

Multiple groups can have the same name but always have a unique ObjectId. When more than one group that has the same name is returned, use the ObjectId in the output

to identify the one you want to use. You then use the output of this command with Set-AzKeyVaultAccessPolicy to grant permissions to group2 for your key vault, named

myownvault . This example enumerates the groups named 'group2' inline in the same command line. There may be multiple groups in the returned list that are named

'group2'. This example picks the first one, indicated by index [0] in the returned list.

Example 7: Grant Azure Information Protection access to the customer-managed tenant key (BYOK)

Set-AzKeyVaultAccessPolicy -VaultName 'Contoso04Vault' -ServicePrincipalName 'MyServicePrincipal' -PermissionsToKeys decrypt,sign,get

This command authorizes Azure Information Protection to use a customer-managed key (the bring your own key, or "BYOK" scenario) as the Azure Information Protection

tenant key. When you run this command, specify your own key vault name but you must specify the ServicePrincipalName parameter and specify the permissions in the

example.

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/az.keyvault/set-azkeyvaultaccesspolicy

Get-AzKeyVault

Remove-AzKeyVaultAccessPolicy