Windows PowerShell Get-Help on Cmdlet 'Set-AzKeyVaultSecret'

PS:\>Get-HELP Set-AzKeyVaultSecret -Full

NAME

Set-AzKeyVaultSecret

SYNOPSIS

Creates or updates a secret in a key vault.

SYNTAX

Set-AzKeyVaultSecret [-InputObject] <Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultSecretIdentityItem> [-SecretValue] <System.Security.SecureString> [-ContentType <System.String>] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Disable] [-Expires <System.Nullable`1[System.DateTime]>] [-NotBefore <System.Nullable`1[System.DateTime]>] [-Tag <System.Collections.Hashtable>] [-Confirm] [-WhatIf] [<CommonParameters>]

Set-AzKeyVaultSecret [-VaultName] <System.String> [-Name] <System.String> [-SecretValue] <System.Security.SecureString> [-ContentType <System.String>] [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-Disable] [-Expires <System.Nullable`1[System.DateTime]>] [-NotBefore <System.Nullable`1[System.DateTime]>] [-Tag <System.Collections.Hashtable>] [-Confirm] [-WhatIf]

[<CommonParameters>]

DESCRIPTION

The Set-AzKeyVaultSecret cmdlet creates or updates a secret in a key vault in Azure Key Vault. If the secret does not exist, this cmdlet creates it. If the secret

already exists, this cmdlet creates a new version of that secret.

PARAMETERS

-ContentType <System.String>

Specifies the content type of a secret. To delete the existing content type, specify an empty string.

Required? false

Position? named

Default value None

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

-Disable <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet disables a secret.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Expires <System.Nullable`1[System.DateTime]>

Specifies the expiration time, as a DateTime object, for the secret that this cmdlet updates. This parameter uses Coordinated Universal Time (UTC). To obtain a

DateTime object, use the Get-Date cmdlet. For more information, type `Get-Help Get-Date`.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

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

Secret object

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-Name <System.String>

Specifies the name of a secret to modify. This cmdlet constructs the fully qualified domain name (FQDN) of a secret based on the name that this parameter

specifies, the name of the key vault, and your current environment.

Required? true

Position? 1

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-NotBefore <System.Nullable`1[System.DateTime]>

Specifies the time, as a DateTime object, before which the secret cannot be used. This parameter uses UTC. To obtain a DateTime object, use the Get-Date cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SecretValue <System.Security.SecureString>

Specifies the value for the secret as a SecureString object. To obtain a SecureString object, use the ConvertTo-SecureString cmdlet. For more information, type

`Get-Help ConvertTo-SecureString`.

Required? true

Position? 2

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Tag <System.Collections.Hashtable>

Key-value pairs in the form of a hash table. For example: @{key0="value0";key1=$null;key2="value2"}

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-VaultName <System.String>

Specifies the name of the key vault to which this secret belongs. This cmdlet constructs the FQDN of a key vault based on the name that this parameter specifies

and your current environment.

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.PSKeyVaultSecretIdentityItem

OUTPUTS

Microsoft.Azure.Commands.KeyVault.Models.PSKeyVaultSecret

NOTES

Example 1: Modify the value of a secret using default attributes

```
$Secret = ConvertTo-SecureString -String 'Password' -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName 'Contoso' -Name 'ITSecret' -SecretValue $Secret

Vault Name   : Contoso
Name         : ITSecret
Version      : 8b5c0cb0326e4350bd78200fac932b51
Id           : https://contoso.vault.azure.net:443/secrets/ITSecret/8b5c0cb0326e4350bd78200fac932b51
Enabled      : True
Expires      :
Not Before   :
Created      : 5/25/2018 6:39:30 PM
Updated      : 5/25/2018 6:39:30 PM
Content Type :
Tags         :
```

The first command converts a string into a secure string by using the ConvertTo-SecureString cmdlet, and then stores that string in the $Secret variable. For more

information, type `Get-Help ConvertTo-SecureString`. The second command modifies value of the secret named ITSecret in the key vault named Contoso. The secret value

becomes the value stored in $Secret.

Example 2: Modify the value of a secret using custom attributes

```
$Secret = ConvertTo-SecureString -String 'Password' -AsPlainText -Force
$Expires = (Get-Date).AddYears(2).ToUniversalTime()
$NBF =(Get-Date).ToUniversalTime()
$Tags = @{ 'Severity' = 'medium'; 'IT' = 'true'}
$ContentType = 'txt'
Set-AzKeyVaultSecret -VaultName 'Contoso' -Name 'ITSecret' -SecretValue $Secret -Expires $Expires -NotBefore $NBF -ContentType $ContentType -Disable -Tags $Tags

Vault Name   : Contoso
Name         : ITSecret
Version      : a2c150be3ea24dd6b8286986e6364851
Id           : https://contoso.vault.azure.net:443/secrets/ITSecret/a2c150be3ea24dd6b8286986e6364851
Enabled      : False
Expires      : 5/25/2020 6:40:00 PM
Not Before   : 5/25/2018 6:40:05 PM
Created      : 5/25/2018 6:41:22 PM
Updated      : 5/25/2018 6:41:22 PM
Content Type : txt
Tags         : Name      Value
               Severity  medium
               IT        true
```

The first command converts a string into a secure string by using the ConvertTo-SecureString cmdlet, and then stores that string in the $Secret variable. For more

information, type `Get-Help ConvertTo-SecureString`. The next commands define custom attributes for the expiry date, tags, and context type, and store the attributes

in variables. The final command modifies values of the secret named ITSecret in the key vault named Contoso, by using the values specified previously as variables.

Example 3: Create a secret in azure key vault by command Set-Secret in module Microsoft.PowerShell.SecretManagement

```
# Install module Microsoft.PowerShell.SecretManagement
Install-Module Microsoft.PowerShell.SecretManagement -Repository PSGallery -AllowPrerelease
# Register vault for Secret Management
 Register-SecretVault -Name AzKeyVault -ModuleName Az.KeyVault -VaultParameters @{ AZKVaultName = 'test-kv'; SubscriptionId = 'xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx' }
# Set secret for vault AzKeyVault
$secure = ConvertTo-SecureString -String "Password" -AsPlainText -Force
Set-Secret -Name secureSecret -SecureStringSecret $secure -Vault AzKeyVault
```

None

This example sets a secret named `secureSecret` in azure key vault `test-kv` by command `Set-Secret` in module `Microsoft.PowerShell.SecretManagement`.

RELATED LINKS

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

Get-AzKeyVaultSecret

Remove-AzKeyVaultSecret