



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

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

### NAME

Set-SqlCredential

### SYNOPSIS

Sets the properties for the SQL Credential object.

### SYNTAX

```
Set-SqlCredential [-InputObject] <Credential> [-Identity] <String> [[-Secret] <SecureString>] [-AccessToken <PSObject>]
[-Encrypt {Mandatory | Optional | Strict}]
[-HostNameInCertificate <String>] [-ProgressAction <ActionPreference>] [-Script] [-TrustServerCertificate] [-Confirm]
[-WhatIf] [<CommonParameters>]
```

```
Set-SqlCredential [[-Path] <String>] [-Identity] <String> [[-Secret] <SecureString>] [-AccessToken <PSObject>] [-Encrypt
{Mandatory | Optional | Strict}]
[-HostNameInCertificate <String>] [-ProgressAction <ActionPreference>] [-Script] [-TrustServerCertificate] [-Confirm]
[-WhatIf] [<CommonParameters>]
```

### DESCRIPTION

The Set-SqlCredential cmdlet sets the Identity and password properties for a SQL Credential object using this cmdlet. This cmdlet supports the two following modes of operation:

- Specify the path or location of the credential including the credential name that uses the Path parameter.
- Pass a Smo.Credential object to the object that uses the InputObject parameter.

## PARAMETERS

-AccessToken <PSObject>

The access token used to authenticate to SQL Server, as an alternative to user/password or Windows Authentication.

This can be used, for example, to connect to `SQL Azure DB` and `SQL Azure Managed Instance` using a `Service Principal` or a `Managed Identity`.

The parameter to use can be either a string representing the token or a `PSAccessToken` object as returned by running `Get-AzAccessToken -ResourceUri

`https://database.windows.net`.`

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Encrypt <String>

The encryption type to use when connecting to SQL Server.

This value maps to the `Encrypt` property `SqlConnectionEncryptOption` on the SqlConnection object of the Microsoft.Data.SqlClient driver.

In v22 of the module, the default is 'Optional' (for compatibility with v21). In v23+ of the module, the default value will be 'Mandatory', which may create a breaking change for existing scripts.

> This parameter is new in v22 of the module.

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

-HostNameInCertificate <String>

The host name to be used in validating the SQL Server TLS/SSL certificate. You must pass this parameter if your SQL Server instance is enabled for Force

Encryption and you want to connect to an instance using hostname/shortname. If this parameter is omitted then passing the Fully Qualified Domain Name (FQDN) to

-ServerInstance is necessary to connect to a SQL Server instance enabled for Force Encryption.

> This parameter is new in v22 of the module.

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

-Identity <String>

Specifies the user or account name for the resource SQL Server needs to authenticate to. For Windows Azure storage service, this is the name of the Windows Azure storage account.

Required?	true
Position?	2
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### **-InputObject <Credential>**

Specifies an input Credential object. To get this object, use the Get-SqlCredential cmdlet.

Required?	true
Position?	1
Default value	None
Accept pipeline input?	True (ByValue)
Accept wildcard characters?	false

#### **-Path <String>**

Specifies the path to the credential on which this cmdlet performs this operation. For instance, ``SQLSERVER:\SQL\Computer\Instance\Credentials\Credential``.

Required?	false
Position?	1
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### **-ProgressAction <ActionPreference>**

Determines how PowerShell responds to progress updates generated by a script, cmdlet, or provider, such as the progress bars generated by the Write-Progress cmdlet. The Write-Progress cmdlet creates progress bars that show a command's status.

Required?	false
Position?	named
Default value	None

Accept pipeline input? False

Accept wildcard characters? false

#### **-Script [<SwitchParameter>]**

Indicates that this cmdlet returns a Transact-SQL script that performs the task that this cmdlet performs.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

#### **-Secret <SecureString>**

Specifies the password for the user or account. For Windows Azure storage service, this is the access key value for the Windows Azure storage account.

Required? false

Position? 3

Default value None

Accept pipeline input? False

Accept wildcard characters? false

#### **-TrustServerCertificate [<SwitchParameter>]**

Indicates whether the channel will be encrypted while bypassing walking the certificate chain to validate trust.

In v22 of the module, the default is ``$true`` (for compatibility with v21). In v23+ of the module, the default value will be `'$false'`, which may create a breaking change for existing scripts.

> This parameter is new in v22 of the module.

Required? false

Position? named

Default value            False  
Accept pipeline input?    False  
Accept wildcard characters? false

**-Confirm [<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 [<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.SqlServer.Management.Smo.Credential

## OUTPUTS

## NOTES

---- Example 1: Set the identity of a SQL credential object ----

```
PS C:\> Set-SqlCredential -Path SQLSERVER:\SQL\Computer\Instance\Credentials\MySqlCredential -Identity "MyStorageAccount"
```

This command sets the identity of `MySqlCredential` to `MyStorageAccount` by specifying the path of the SQL Credential.

Example 2: Set the identity of a SQL credential object using the pipeline

```
PS C:\> $Cred = Get-SqlCredential -Name "MySqlCredential"
PS C:\> $Cred | Set-SqlCredential -Identity "MyStorageAccount"
```

The first command gets the credential object from the Get-Credential cmdlet and stores the result in the variable named `\$Cred`.

Example 3: Set the identity of a SQL credential object prompting the user

```
PS C:\> $Secret = Read-Host "Please enter the storage account access key"
PS C:\> Set-SqlCredential -Identity "MyStorageAccount" -Secret $Secret
```

The first command prompts for the storage access key information and stores the result in the variable named `\$Secret`.

## RELATED LINKS

Online Version: <https://learn.microsoft.com/powershell/module/sqlserver/set-sqlcredential>

Get-SqlCredential