



python



PowerShell

FPDF Library  
PDF generator

*Full credit is given to the above companies including the OS that this PDF file was generated!*

### **PowerShell Get-Help on command 'Set-LocalUser'**

**PS C:\Users\wahid> Get-Help Set-LocalUser**

#### NAME

Set-LocalUser

#### SYNOPSIS

Modifies a local user account.

#### SYNTAX

```
Set-LocalUser [-InputObject] <Microsoft.PowerShell.Commands.LocalUser>  
[-AccountExpires <System.DateTime>] [-AccountNeverExpires] [-Description  
<System.String>] [-FullName <System.String>] [-Password  
<System.Security.SecureString>] [-PasswordNeverExpires <System.Boolean>]  
[-UserMayChangePassword <System.Boolean>] [-Confirm] [-WhatIf]  
[<CommonParameters>]
```

```
Set-LocalUser [-Name] <System.String> [-AccountExpires <System.DateTime>]  
[-AccountNeverExpires] [-Description <System.String>] [-FullName  
<System.String>] [-Password <System.Security.SecureString>]  
[-PasswordNeverExpires <System.Boolean>] [-UserMayChangePassword  
<System.Boolean>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

```
Set-LocalUser [-SID] <System.Security.Principal.SecurityIdentifier>
[-AccountExpires <System.DateTime>] [-AccountNeverExpires] [-Description
<System.String>] [-FullName <System.String>] [-Password
<System.Security.SecureString>] [-PasswordNeverExpires <System.Boolean>]
[-UserMayChangePassword <System.Boolean>] [-Confirm] [-WhatIf]
[<CommonParameters>]
```

## DESCRIPTION

The `Set-LocalUser` cmdlet modifies a local user account. This cmdlet can reset the password of a local user account.

> [!NOTE] > The `Microsoft.PowerShell.LocalAccounts` module is not available in 32-bit PowerShell on a 64-bit > system.

## PARAMETERS

`-AccountExpires <System.DateTime>`

Specifies when the user account expires. To obtain a `DateTime` object, use the `Get-Date` cmdlet.

If you do not want the account to expire, specify the `AccountNeverExpires` parameter.

`-AccountNeverExpires <System.Management.Automation.SwitchParameter>`

Indicates that the account does not expire.

`-Description <System.String>`

Specifies a comment for the user account. The maximum length is 48 characters.

`-FullName <System.String>`

Specifies the full name for the user account. The full name differs from

the user name of the user account.

`-InputObject <Microsoft.PowerShell.Commands.LocalUser>`

Specifies the user account that this cmdlet changes. To obtain a user account, use the ``Get-LocalUser`` cmdlet.

`-Name <System.String>`

Specifies the name of the user account that this cmdlet changes.

`-Password <System.Security.SecureString>`

Specifies a password for the user account. If the user account is connected to a Microsoft account, do not set a password.

You can use ``Read-Host -AsSecureString``, ``Get-Credential``, or ``ConvertTo-SecureString`` to create a `SecureString` object for the password.

If you omit the `Password` and `NoPassword` parameters, ``Set-LocalUser`` prompts you for the user's password.

`-PasswordNeverExpires <System.Boolean>`

Indicates whether the password expires.

`-SID <System.Security.Principal.SecurityIdentifier>`

Specifies the security ID (SID) of the user account that this cmdlet changes.

`-UserMayChangePassword <System.Boolean>`

Indicates that the user can change the password on the user account.

`-Confirm <System.Management.Automation.SwitchParameter>`

Prompts you for confirmation before running the cmdlet.

`-WhatIf <System.Management.Automation.SwitchParameter>`

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

#### <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>).

----- Example 1: Change a description of a user account -----

```
Set-LocalUser -Name "Admin07" -Description "Description of this account."
```

This command changes the description of a user account named Admin07.

----- Example 2: Change the password on an account -----

```
$Password = Read-Host -AsSecureString  
$UserAccount = Get-LocalUser -Name "User02"  
$UserAccount | Set-LocalUser -Password $Password
```

The first command prompts you for a password by using the Read-Host cmdlet.

The command stores the password as a secure string in the \$Password variable.

The second command gets a user account named User02 by using `Get-LocalUser`.

The command stores the account in the \$UserAccount variable.

The third command sets the new password on the user account stored in

`\$UserAccount`.

#### REMARKS

To see the examples, type: "get-help Set-LocalUser -examples".

For more information, type: "get-help Set-LocalUser -detailed".

For technical information, type: "get-help Set-LocalUser -full".

For online help, type: "get-help Set-LocalUser -online"

