



Full credit is given to all the above companies including the Operating System that this PDF file was generated!

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

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

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.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-AccountNeverExpires <System.Management.Automation.SwitchParameter>

Indicates that the account does not expire.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Description <System.String>

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

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-FullName <System.String>

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

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-InputObject <Microsoft.PowerShell.Commands.LocalUser>

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

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName, ByValue)

Accept wildcard characters? false

-Name <System.String>

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

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName, ByValue)

Accept wildcard characters? false

-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.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-PasswordNeverExpires <System.Boolean>

Indicates whether the password expires.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-SID <System.Security.Principal.SecurityIdentifier>

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

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName, ByValue)

Accept wildcard characters? false

-UserMayChangePassword <System.Boolean>

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

Required? false

Position? named

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

`System.Management.Automation.SecurityAccountsManager.LocalUser`

You can pipe a local user to this cmdlet.

`System.String`

You can pipe a string to this cmdlet.

`System.Security.Principal.SecurityIdentifier`

You can pipe a SID to this cmdlet.

OUTPUTS

`None`

This cmdlet returns no output.

NOTES

Windows PowerShell includes the following aliases for `Set-LocalUser`:

- `slu`

The `PrincipalSource` property is a property on `LocalUser` , `LocalGroup` , and `LocalPrincipal` objects that describes the source of the object. The possible sources

are as follows:

- Local

- Active Directory

- Microsoft Entra group

- Microsoft Account PrincipalSource is supported only by Windows 10, Windows Server 2016, and later versions of the Windows operating system. For earlier versions, the property is blank.

----- 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`.

RELATED LINKS

	Online	Version:
https://learn.microsoft.com/powershell/module/microsoft.powershell.localaccounts/set-localuser?view=powershell-5.1&WT.mc_id=ps-gethelp		
Disable-LocalUser		
Enable-LocalUser		
Get-LocalUser		

New-LocalUser

Remove-LocalUser

Rename-LocalUser