

Download PDF

```
Learn / PowerShell / Microsoft.PowerShell.Utility /
Unblock-File
                                                                               Feedback
Reference
Module: Microsoft.PowerShell.Utility
In this article
  Syntax
  Description
  Examples
  Parameters
```

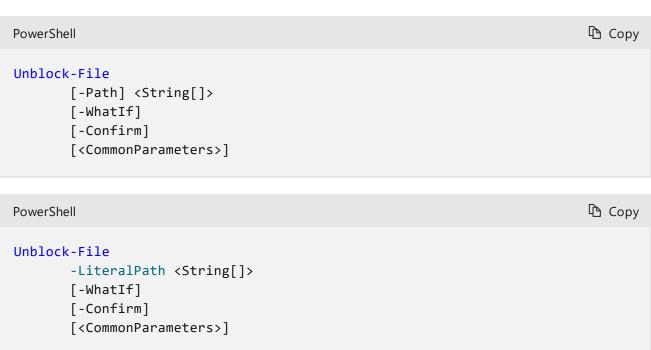
Sign in

**Download PowerShell** 

Unblocks files that were downloaded from the internet.

## **Syntax**

Show 4 more



## Description

This cmdlet only works on the Windows and macOS platforms.

The Unblock-File cmdlet lets you open files that were downloaded from the internet. It unblocks PowerShell script files that were downloaded from the internet so you can run them, even when the PowerShell execution policy is RemoteSigned. By default, these files are blocked to protect the computer from untrusted files.

Before using the Unblock-File cmdlet, review the file and its source and verify that it is safe to open.

Internally, the Unblock-File cmdlet removes the **Zone.Identifier** alternate data stream, which has a value of 3 to indicate that it was downloaded from the internet.

For more information about PowerShell execution policies, see about\_Execution\_Policies.

This cmdlet was introduced in Windows PowerShell 3.0.

## **Examples**

### Example 1: Unblock a file

This command unblocks the PowerShellTips.chm file.

```
PowerShell

PS C:\> Unblock-File -Path C:\Users\User01\Documents\Downloads\PowerShellTips.ch
```

### **Example 2: Unblock multiple files**

This command unblocks all of the files in the C:\Downloads directory whose names include "PowerShell". Do not run a command like this one until you have verified that all files are safe.

```
PowerShell

PS C:\> dir C:\Downloads\*PowerShell* | Unblock-File
```

### Example 3: Find and unblock scripts

This command shows how to find and unblock PowerShell scripts.

The first command uses the **Stream** parameter of the *Get-Item* cmdlet get files with the **Zone.Identifier** stream.

The second command shows what happens when you run a blocked script in a PowerShell session in which the execution policy is **RemoteSigned**. The **RemoteSigned** policy prevents you from running scripts that are downloaded from the internet unless they are digitally signed.

The third command uses the Unblock-File cmdlet to unblock the script so it can run in the session.

```
Copy
PowerShell
PS C:\> Get-Item * -Stream "Zone.Identifier" -ErrorAction SilentlyContinue
   FileName: C:\ps-test\Start-ActivityTracker.ps1
                       Length
Stream
Zone.Identifier
PS C:\> C:\ps-test\Start-ActivityTracker.ps1
c:\ps-test\Start-ActivityTracker.ps1 : File c:\ps-test\Start-ActivityTracker.ps1
be loaded. The file c:\ps-test\Start-ActivityTracker.ps1 is not digitally signed
will not execute on the system. For more information, see about_Execution_Polici
At line:1 char:1
+ c:\ps-test\Start-ActivityTracker.ps1
    + CategoryInfo
                           : SecurityError: (:) [], PSSecurityException
   + FullyQualifiedErrorId : UnauthorizedAccess
PS C:\> Get-Item C:\ps-test\Start-ActivityTracker.ps1 | Unblock-File
```

### **Parameters**

-Confirm

Prompts you for confirmation before running the cmdlet.

Expand table	٦.	Expand	table
--------------	----	--------	-------

Туре:	SwitchParameter
Aliases:	cf
Position:	Named
Default value:	False
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

#### -LiteralPath

Specifies the files to unblock. Unlike **Path**, the value of the **LiteralPath** parameter is used exactly as it is typed. No characters are interpreted as wildcards. If the path includes escape characters, enclose it in single quotation marks (•). Single quotation marks tell PowerShell not to interpret any characters as escape sequences.

### **Expand table**

Туре:	String[]
Aliases:	PSPath, LP
Position:	Named
Default value:	None
Required:	True
Accept pipeline input:	True
Accept wildcard characters:	False

#### -Path

Specifies the files to unblock. Wildcard characters are supported.

### **Expand table**

Type:	String[]
Position:	0
Default value:	None
Required:	True
Accept pipeline input:	False
Accept wildcard characters:	True

#### -WhatIf

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

### **Expand table**

Туре:	SwitchParameter
Aliases:	wi
Position:	Named

Default value:	False
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

## Inputs

#### **String**

You can pipe a file path to this cmdlet.

## **Outputs**

#### None

This cmdlet returns no output.

### **Notes**

- Support for macOS was added in PowerShell 7.
- The Unblock-File cmdlet works only in file system drives.
- Unblock-File performs the same operation as the **Unblock** button on the **Properties** dialog box in File Explorer.
- If you use the Unblock-File cmdlet on a file that is not blocked, the command has no effect on the unblocked file and the cmdlet does not generate errors.

### **Related Links**

- about\_Execution\_Policies
- Get-Item
- Out-File
- FileSystem Provider

# Collaborate with us on GitHub

The source for this content can be found on GitHub, where you can also create and review issues and pull requests. For more information, see our contributor guide.



#### PowerShell feedback

PowerShell is an open source project. Select a link to provide feedback:

🖔 Open a documentation issue

Provide product feedback

S English (United States)

**✓**× Your Privacy Choices

☆ Theme ∨

Manage cookies Previous Versions

Blog ☑ Contribute

Privacy ☑

Terms of Use

Trademarks ☑

© Microsoft 2024