



**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 'Get-ChildItem'***

**PS:\>Get-HELP Get-ChildItem -Full**

#### **NAME**

Get-ChildItem

#### **SYNOPSIS**

Gets the items and child items in one or more specified locations.

#### **SYNTAX**

```
Get-ChildItem [[-Filter] <System.String>] [-Attributes {Archive | Compressed | Device | Directory | Encrypted | Hidden | IntegrityStream | Normal | NoScrubData | NotContentIndexed | Offline | ReadOnly | ReparsePoint | SparseFile | System | Temporary}] [-CodeSigningCert] [-Depth <System.UInt32>] [-Directory] [-DnsName <Microsoft.PowerShell.Commands.DnsNameRepresentation>] [-DocumentEncryptionCert] [-Eku <System.String>] [-Exclude <System.String[]>] [-ExpiringInDays <System.Int32>] [-File] [-FollowSymlink] [-Force] [-Hidden] [-Include <System.String[]>] [-LiteralPath <System.String[]>] [-Name] [-ReadOnly] [-Recurse] [-SSLServerAuthentication] [-System] [-UseTransaction] [<CommonParameters>]
```

```
Get-ChildItem [[-Path] <System.String[]>] [[-Filter] <System.String>] [-Attributes {Archive | Compressed | Device | Directory | Encrypted | Hidden | IntegrityStream |
```

Normal | NoScrubData | NotContentIndexed | Offline | ReadOnly | ReparsePoint | SparseFile | System | Temporary}] [-CodeSigningCert] [-Depth <System.UInt32>]  
[-Directory] [-DnsName <Microsoft.PowerShell.Commands.DnsNameRepresentation>] [-DocumentEncryptionCert] [-Eku <System.String>] [-Exclude <System.String[]>]  
[-ExpiringInDays <System.Int32>] [-File] [-FollowSymlink] [-Force] [-Hidden] [-Include <System.String[]>] [-Name]  
[-ReadOnly] [-Recurse] [-SSLServerAuthentication]  
[-System] [-UseTransaction] [<CommonParameters>]

## DESCRIPTION

The `Get-ChildItem` cmdlet gets the items in one or more specified locations. If the item is a container, it gets the items inside the container, known as child

items. You can use the Recurse parameter to get items in all child containers and use the Depth parameter to limit the number of levels to recurse.

`Get-ChildItem` doesn't display empty directories. When a `Get-ChildItem` command includes the Depth or Recurse parameters, empty directories aren't included in the output.

Locations are exposed to `Get-ChildItem` by PowerShell providers. A location can be a file system directory, registry hive, or a certificate store. Some parameters

are only available for a specific provider. For more information, see [about\\_Providers](#) (./Microsoft.PowerShell.Core/About/about\_Providers.md).

## PARAMETERS

-Attributes <System.Management.Automation.FlagsExpression`1[System.IO.FileAttributes]>  
> [!NOTE] > This parameter is only available in the > FileSystem  
(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

Gets files and folders with the specified attributes. This parameter supports all attributes and lets you specify complex combinations of attributes.

For example, to get non-system files (not directories) that are encrypted or compressed, type:

```
`Get-ChildItem -Attributes !Directory+!System+Encrypted, !Directory+!System+Compressed`
```

To find files and folders with commonly used attributes, use the Attributes parameter. Or, the parameters Directory , File , Hidden , ReadOnly , and System .

The Attributes parameter supports the following properties:

- Archive - Compressed - Device - Directory - Encrypted - Hidden - IntegrityStream - Normal - NoScrubData - NotContentIndexed - Offline - ReadOnly - ReparsePoint

- SparseFile - System - Temporary For a description of these attributes, see the `FileAttributes` Enumeration ([/dotnet/api/system.io.fileattributes](#)).

To combine attributes, use the following operators:

- `!` (NOT)

- `+` (AND)

- `,` (OR)

Don't use spaces between an operator and its attribute. Spaces are accepted after commas.

For common attributes, use the following abbreviations:

- `D` (Directory)

- `H` (Hidden)

- `R` (Read-only)

- `S` (System)

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

-CodeSigningCert <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > Certificate  
(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

To get a list of certificates that have `Code Signing` in their EnhancedKeyUsageList property value, use the CodeSigningCert parameter.

Required? false  
Position? named  
Default value False  
Accept pipeline input? False  
Accept wildcard characters? false

-Depth <System.UInt32>

This parameter was added in PowerShell 5.0 and enables you to control the depth of recursion. By default, `Get-ChildItem` displays the contents of the parent directory. The Depth parameter determines the number of subdirectory levels that are included in the recursion and displays the contents.

For example, `-Depth 2` includes the Path parameter's directory, first level of subdirectories, and second level of subdirectories. By default directory names and filenames are included in the output.

> [!NOTE] > On a Windows computer from PowerShell or cmd.exe , you can display a graphical view of a > directory structure with the tree.com command.

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

-Directory <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem  
(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

To get a list of directories, use the Directory parameter or the Attributes parameter with the Directory property. You can use the Recurse parameter with

Directory .

Required? false  
Position? named  
Default value False  
Accept pipeline input? False  
Accept wildcard characters? false

-DnsName <Microsoft.PowerShell.Commands.DnsNameRepresentation>

> [!NOTE] > This parameter is only available in the > Certificate  
(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

Specifies a domain name or name pattern to match with the DnsNameList property of certificates the cmdlet gets. The value of this parameter can either be

`Unicode` or `ASCII`. Punycode values are converted to Unicode. Wildcard characters (`\*`) are permitted.

This parameter was reintroduced in PowerShell 7.1

Required? false  
Position? named

Default value                None

Accept pipeline input?    False

Accept wildcard characters? true

-DocumentEncryptionCert <System.Management.Automation.SwitchParameter>

    > [!NOTE] > This parameter is only available in the > Certificate

(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

To get a list of certificates that have `Document Encryption` in their EnhancedKeyUsageList property value, use the DocumentEncryptionCert parameter.

Required?                false

Position?                named

Default value                False

Accept pipeline input?    False

Accept wildcard characters? false

-Eku <System.String>

    > [!NOTE] > This parameter is only available in the > Certificate

(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

Specifies text or a text pattern to match with the EnhancedKeyUsageList property of certificates the cmdlet gets. Wildcard characters (^ ~) are permitted. The

EnhancedKeyUsageList \* property contains the friendly name and the OID fields of the EKU.

This parameter was reintroduced in PowerShell 7.1

Required?                false

Position?                named

Default value                None

Accept pipeline input?    False

Accept wildcard characters? true

#### -Exclude <System.String[]>

Specifies an array of one or more string patterns to be matched as the cmdlet gets child items. Any matching item is excluded from the output. Enter a path

element or pattern, such as ` .txt` or ` A `. Wildcard characters are accepted.

A trailing asterisk ( ` ) in the Path \* parameter is optional. For example, `-Path C:\Test\Logs` or `-Path C:\Test\Logs\ `. If a trailing asterisk ( ` ) is

included, the command recurses into the Path parameter's subdirectories. Without the asterisk ( ` ), the contents of the Path \* parameter are displayed. More

details are included in Example 5 and the Notes section.

The Include and Exclude parameters can be used together. However, the exclusions are applied after the inclusions, which can affect the final output.

> [!NOTE] > The Include and Exclude parameters have no effect when used with the LiteralPath > parameter. This is fixed in PowerShell 7.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? true

#### -ExpiringInDays <System.Int32>

> [!NOTE] > This parameter is only available in the > Certificate  
(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

Specifies that the cmdlet should only return certificates that are expiring in or before the specified number of days. A value of zero (`0`) gets certificates that have expired.

This parameter was reintroduced in PowerShell 7.1

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

-File <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem  
(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

To get a list of files, use the File parameter. You can use the Recurse parameter with File .

Required? false  
Position? named  
Default value False  
Accept pipeline input? False  
Accept wildcard characters? false

-Filter <System.String>

Specifies a filter to qualify the Path parameter. The FileSystem  
(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider is the only installed

PowerShell provider that supports filters. Filters are more efficient than other parameters. The provider applies filter  
when the cmdlet gets the objects rather

than having PowerShell filter the objects after they're retrieved. The filter string is passed to the .NET API to enumerate  
files. The API only supports `\*` and

`?` wildcards.

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

-FollowSymlink <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem

(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

By default, the `Get-ChildItem` cmdlet displays symbolic links to directories found during recursion, but doesn't recurse into them. Use the FollowSymlink

parameter to search the directories that target those symbolic links. The FollowSymlink is a dynamic parameter and is supported only in the FileSystem provider.

This parameter was introduced in PowerShell 6.0.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Force <System.Management.Automation.SwitchParameter>

Allows the cmdlet to get items that otherwise can't be accessed by the user, such as hidden or system files. The Force parameter doesn't override security

restrictions. Implementation varies among providers. For more information, see about\_Providers  
(./Microsoft.PowerShell.Core/About/about\_Providers.md).

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Hidden <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem

(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

To get only hidden items, use the Hidden parameter or the Attributes parameter with the Hidden property. By default, `Get-ChildItem` doesn't display hidden items.

Use the Force parameter to get hidden items.

Required?	false
Position?	named
Default value	False
Accept pipeline input?	False
Accept wildcard characters?	false

#### -Include <System.String[]>

Specifies an array of one or more string patterns to be matched as the cmdlet gets child items. Any matching item is included in the output. Enter a path element

or pattern, such as `"\* .txt"`. Wildcard characters are permitted. The Include parameter is effective only when the command includes the contents of an item, such

as `C:\Windows\*`, where the wildcard character specifies the contents of the `C:\Windows` directory.

The Include and Exclude parameters can be used together. However, the exclusions are applied after the inclusions, which can affect the final output.

> [!NOTE] > The Include and Exclude parameters have no effect when used with the LiteralPath > parameter. This is fixed in PowerShell 7.

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

#### -LiteralPath <System.String[]>

Specifies a path to one or more locations. The value of LiteralPath is used exactly as it's typed. No characters are interpreted as wildcards. If the path

includes escape characters, enclose it in single quotation marks. Single quotation marks tell PowerShell to interpret

any characters as escape sequences.

For more information, see [about\\_Quoting\\_Rules](#) (./Microsoft.PowerShell.Core/About/about\_Quoting\_Rules.md).

> [!NOTE] > The Include and Exclude parameters have no effect when used with the LiteralPath > parameter. This is fixed in PowerShell 7.

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

#### -Name <System.Management.Automation.SwitchParameter>

Gets only the names of the items in the location. The output is a string object that can be sent down the pipeline to other commands. The names returned are

relative to the value of the Path parameter.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

#### -Path <System.String[]>

Specifies a path to one or more locations. Wildcards are accepted. The default location is the current directory ('.') .

Required? false

Position? 0

Default value Current directory

Accept pipeline input? True (ByPropertyName, ByValue)

Accept wildcard characters? true

-ReadOnly <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem

(./Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

To get only read-only items, use the `ReadOnly` parameter or the `Attributes` parameter `ReadOnly` property.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-Recurse <System.Management.Automation.SwitchParameter>

Gets the items in the specified locations and in all child items of the locations.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-SSLServerAuthentication <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > Certificate

(./Microsoft.PowerShell.Security/About/about\_Certificate\_Provider.md)provider.

To get a list of certificates that have `Server Authentication` in their `EnhancedKeyUsageList` property value, use the `SSLServerAuthentication` parameter.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-System <System.Management.Automation.SwitchParameter>

> [!NOTE] > This parameter is only available in the > FileSystem

(../Microsoft.PowerShell.Core/About/about\_FileSystem\_Provider.md)provider.

Gets only system files and directories. To get only system files and folders, use the System parameter or Attributes parameter System property.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-UseTransaction <System.Management.Automation.SwitchParameter>

Includes the command in the active transaction. This parameter is valid only when a transaction is in progress. For more information, see about\_Transactions

(../Microsoft.PowerShell.Core/About/about\_Transactions.md).

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.String

Page 13/23

You can pipe a string that contains a path to this cmdlet.

## OUTPUTS

`System.Management.Automation.AliasInfo`

The cmdlet outputs this type when accessing the `Alias:` drive.

`Microsoft.PowerShell.Commands.X509StoreLocation`

`System.Security.Cryptography.X509Certificates.X509Store`

`System.Security.Cryptography.X509Certificates.X509Certificate2`

The cmdlet outputs these types when accessing the `Cert:` drive.

`System.Collections.DictionaryEntry`

The cmdlet outputs this type when accessing the `Env:` drive.

`System.IO.DirectoryInfo`

`System.IO.FileInfo`

The cmdlet outputs these types when accessing the Filesystem drives.

`System.Management.Automation.FunctionInfo`

`System.Management.Automation.FilterInfo`

The cmdlet outputs these types when accessing the `Function:` drives.

`Microsoft.Win32.RegistryKey`

The cmdlet outputs this type when accessing the Registry drives.

## **System.Management.Automation.PSVariable**

The cmdlet outputs this type when accessing the `Variable:` drives.

## **Microsoft.WSMan.Management.WSManConfigContainerElement**

## **Microsoft.WSMan.Management.WSManConfigLeafElement**

The cmdlet outputs these types when accessing the `WSMan:` drives.

## **System.String**

When you use the Name parameter, this cmdlet returns the object names as strings.

## **NOTES**

Windows PowerShell includes the following aliases for `Get-ChildItem`:

- `ls`

- `dir`

- `gci`

`Get-ChildItem` doesn't get hidden items by default. To get hidden items, use the Force parameter.

The `Get-ChildItem` cmdlet is designed to work with the data exposed by any provider. To list the providers available in your session, type `Get-PSPrinter`. For

more information, see [about\\_Providers \(./Microsoft.PowerShell.Core/About/about\\_Providers.md\)](#).

```
Get-ChildItem -Path C:\Test
```

Directory: C:\Test

Mode	LastWriteTime	Length	Name
---	-----	-----	
d----	2/15/2019	08:29	Logs
-a---	2/13/2019	08:55	26 anotherfile.txt
-a---	2/12/2019	15:40	118014 Command.txt
-a---	2/1/2019	08:43	183 CreateTestFile.ps1
-ar--	2/12/2019	14:31	27 ReadOnlyFile.txt

By default `Get-ChildItem` lists the mode ( Attributes ), LastWriteTime , file size ( Length ), and the Name of the item. The letters in the Mode property can be

interpreted as follows:

- `l` (link)

- `d` (directory)

- `a` (archive)

- `r` (read-only)

- `h` (hidden)

- `s` (system)

For more information about the mode flags, see [about\\_Filesystem\\_Provider](#) ([./microsoft.powershell.core/about/about\\_filesystem\\_provider.md#attributes-flagsexpression](#)).

----- Example 2: Get child item names in a directory -----

```
Get-ChildItem -Path C:\Test -Name
```

Logs

anotherfile.txt

Command.txt

CreateTestFile.ps1

ReadOnlyFile.txt

Example 3: Get child items in the current directory and subdirectories

```
Get-ChildItem -Path .\*.txt -Recurse -Force
```

Directory: C:\Test\Logs\Adirectory

Mode	LastWriteTime	Length	Name
---	-----	-----	
-a---	2/12/2019 16:16	20	Afile4.txt
-a-h--	2/12/2019 15:52	22	hiddenfile.txt
-a---	2/13/2019 13:26	20	LogFile4.txt

Directory: C:\Test\Logs\Backup

Mode	LastWriteTime	Length	Name
---	-----	-----	
-a---	2/12/2019 16:16	20	ATextField.txt
-a---	2/12/2019 15:50	20	LogFile3.txt

Directory: C:\Test\Logs

Mode	LastWriteTime	Length	Name
------	---------------	--------	------

-a---	2/12/2019	16:16	20 Afile.txt
-a-h--	2/12/2019	15:52	22 hiddenfile.txt
-a---	2/13/2019	13:26	20LogFile1.txt

Directory: C:\Test

Mode	LastWriteTime	Length	Name
-a---	2/13/2019 08:55	26	anotherfile.txt
-a---	2/12/2019 15:40	118014	Command.txt
-a-h--	2/12/2019 15:52	22	hiddenfile.txt
-ar---	2/12/2019 14:31	27	ReadOnlyFile.txt

The `Get-ChildItem` cmdlet uses the Path parameter to specify `C:\Test\\*.txt`. Path \* uses the asterisk (`\*`) wildcard to specify all files with the filename

extension `.txt`. The Recurse \* parameter searches the Path directory its subdirectories, as shown in the Directory: headings. The Force parameter displays hidden files such as `hiddenfile.txt` that have a mode of h .

---- Example 4: Get child items using the Include parameter ----

```
# When using the -Include parameter, if you don't include an asterisk in the path
# the command returns no output.
```

```
Get-ChildItem -Path C:\Test\ -Include *.txt
```

```
Get-ChildItem -Path C:\Test\* -Include *.txt
```

Directory: C:\Test

Mode	LastWriteTime	Length	Name
------	---------------	--------	------

-a----	2/13/2019	08:55	26 anotherfile.txt
-a----	2/12/2019	15:40	118014 Command.txt
-ar---	2/12/2019	14:31	27 ReadOnlyFile.txt

The `Get-ChildItem` cmdlet uses the Path parameter to specify the directory `C:\Test`. The Path parameter includes a trailing asterisk (`\*`) wildcard to specify the directory's contents. The Include parameter uses an asterisk (`\*`) wildcard to specify all files with the file name extension `.txt`.

When the Include parameter is used, the Path parameter needs a trailing asterisk (`\*`) wildcard to specify the directory's contents. For example, `-Path C:\Test\*`.

- If the Recurse parameter is added to the command, the trailing asterisk (`\*`) in the Path \* parameter is optional. The Recurse parameter gets items from the Path

directory and its subdirectories. For example, `-Path C:\Test\ -Recurse -Include \*.txt` - If a trailing asterisk (`\*`) isn't included in the Path \* parameter, the

command doesn't return any output and returns to the PowerShell prompt. For example, `-Path C:\Test`.

---- Example 5: Get child items using the Exclude parameter ----

Get-ChildItem -Path C:\Test\Logs

Directory: C:\Test\Logs

Mode	LastWriteTime	Length	Name
----	-----	-----	
d----	2/15/2019	13:21	Adirectory
d----	2/15/2019	08:28	AnEmptyDirectory
d----	2/15/2019	13:21	Backup
-a----	2/12/2019	16:16	20 Afile.txt
-a----	2/13/2019	13:26	20 LogFile1.txt
-a----	2/12/2019	16:24	23 systemlog1.log

Get-ChildItem -Path C:\Test\Logs\\* -Exclude A\*

Page 19/23

Directory: C:\Test\Logs

Mode	LastWriteTime	Length	Name
---	-----	----	
d----	2/15/2019	13:21	Backup
-a----	2/13/2019	13:26	20LogFile1.txt
-a----	2/12/2019	16:24	23systemlog1.log

The `Get-ChildItem` cmdlet uses the Path parameter to specify the directory `C:\Test\Logs`. The Exclude parameter uses the asterisk (`\*`) wildcard to specify any files or directories that begin with `A` or `a` are excluded from the output.

When the Exclude parameter is used, a trailing asterisk (` `) in the Path \* parameter is optional. For example, `-Path C:\Test\Logs` or `-Path C:\Test\Logs\*`.

- If a trailing asterisk (` `) isn't included in the Path \* parameter, the contents of the Path parameter are displayed. The exceptions are filenames or subdirectory

names that match the Exclude parameter's value. - If a trailing asterisk (` `) is included in the Path \* parameter, the command recurses into the Path parameter's

subdirectories. The exceptions are filenames or subdirectory names that match the Exclude parameter's value. - If the Recurse parameter is added to the command, the

recursion output is the same whether or not the Path parameter includes a trailing asterisk (\*).

---- Example 6: Get the registry keys from a registry hive ----

Get-ChildItem -Path HKLM:\HARDWARE

Hive: HKEY\_LOCAL\_MACHINE\HARDWARE

Name	Property
---	-----
ACPI	

DEVICEMAP

RESOURCEMAP

UEFI

Get-ChildItem -Path HKLM:\HARDWARE -Exclude D\*

Hive: HKEY\_LOCAL\_MACHINE\HARDWARE

Name	Property
------	----------

---	-----
-----	-------

ACPI
------

RESOURCEMAP
-------------

The first command shows the contents of the `HKLM:\HARDWARE` registry key. The Exclude parameter tells `Get-ChildItem` not to return any subkeys that start with `D`.

Currently, the Exclude \* parameter only works on subkeys, not item properties.

- Example 7: Get all certificates with code-signing authority -

Get-ChildItem -Path Cert:\\* -Recurse -CodeSigningCert

For more information about the Certificate provider and the `Cert:` drive, see [about\\_Certificate\\_Provider](#)

([./Microsoft.PowerShell.Security/About/about\\_Certificate\\_Provider.md](#)).

----- Example 8: Get items using the Depth parameter -----

Get-ChildItem -Path C:\Parent -Depth 2

Directory: C:\Parent

Mode	LastWriteTime	Length	Name
---	-----	-----	
d----	2/14/2019	10:24	SubDir_Level1
-a----	2/13/2019	08:55	26 file.txt

Directory: C:\Parent\SubDir\_Level1

Mode	LastWriteTime	Length	Name
---	-----	-----	
d----	2/14/2019 10:24		SubDir_Level2
-a----	2/13/2019 08:55	26	file.txt

Directory: C:\Parent\SubDir\_Level1\SubDir\_Level2

Mode	LastWriteTime	Length	Name
---	-----	-----	
d----	2/14/2019 10:22		SubDir_Level3
-a----	2/13/2019 08:55	26	file.txt

The `Get-ChildItem` cmdlet uses the Path parameter to specify `C:\Parent`. The Depth parameter specifies two levels of recursion. `Get-ChildItem` displays the contents of the directory specified by the Path parameter and the two levels of subdirectories.

----- Example 9: Get the link target for a junction point -----

```
PS D:\> New-Item -ItemType Junction -Name tmp -Target $env:TEMP
```

```
PS D:\> Get-ChildItem | Select-Object name,*target
```

Name	Target
---	-----
tmp	{C:\Users\user1\AppData\Local\Temp}

-- Example 10: Get the link target for an AppX reparse point --

```
Get-ChildItem ~\AppData\Local\Microsoft\WindowsApps\MicrosoftEdge.exe |  
Select-Object Mode, LinkTarget, LinkType, Name
```

Mode LinkTarget LinkType Name

la--- MicrosoftEdge.exe

At this time, Windows doesn't provide a way to get the target information for an AppX reparse point. The LinkTarget and LinkType properties of the filesystem object are empty.

## RELATED LINKS

Online

Version:

[https://learn.microsoft.com/powershell/module/microsoft.powershell.management/get-childitem?view=powershell-5.1&WT.mc\\_id=ps-gethelp](https://learn.microsoft.com/powershell/module/microsoft.powershell.management/get-childitem?view=powershell-5.1&WT.mc_id=ps-gethelp)

[about\\_Certificate\\_Provider](#)

[about\\_Providers](#)

[about\\_Quoting\\_Rules](#)

[about\\_Registry\\_Provider](#)

[ForEach-Object](#)

[Get-Alias](#)

[Get-Item](#)

[Get-Location](#)

[Get-Process](#)

[Get-PSProvider](#)

[Split-Path](#)