

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-Service'

PS:\>Get-HELP Get-Service -Full

NAME

Get-Service

SYNOPSIS

Gets the services on a local or remote computer.

SYNTAX

Get-Service [-ComputerName <System.String[]>] [-DependentServices] -DisplayName <System.String[]> [-Exclude <System.String[]>] [-Include <System.String[]>]

[-RequiredServices] [<CommonParameters>]

Get-Service [-ComputerName <System.String[]>] [-DependentServices] [-Exclude <System.String[]>] [-Include <System.String[]>] [-InputObject

<System.ServiceProcess.ServiceController[]>] [-RequiredServices] [<CommonParameters>]

Get-Service [[-Name] <System.String[]>] [-ComputerName <System.String[]>] [-DependentServices] [-Exclude <System.String[]>] [-Include <System.String[]>]

[-RequiredServices] [<CommonParameters>]

DESCRIPTION

The `Get-Service` cmdlet gets objects that represent the services on a local computer or on a remote computer, including running and stopped services. By default,

when `Get-Service` is run without parameters, all the local computer's services are returned.

You can direct this cmdlet to get only particular services by specifying the service name or the display name of the services, or you can pipe service objects to this cmdlet.

PARAMETERS

-ComputerName <System.String[]>

Gets the services running on the specified computers. The default is the local computer.

Type the NetBIOS name, an IP address, or a fully qualified domain name (FQDN) of a remote computer. To specify the local computer, type the computer name, a dot

(`.`), or `localhost`.

This parameter does not rely on Windows PowerShell remoting. You can use the ComputerName parameter of `Get-Service` even if your computer is not configured to

run remote commands.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

false

Accept wildcard characters? false

-DependentServices <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet gets only the services that depend upon the specified service.

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-DisplayName <System.String[]>

Specifies, as a string array, the display names of services to be retrieved. Wildcards are permitted.

Required? true

Position? named

- Default value None
- Accept pipeline input? False

Accept wildcard characters? true

-Exclude <System.String[]>

Specifies, as a string array, a service or services that this cmdlet excludes from the operation. The value of this parameter qualifies the Name parameter. Enter

a name element or pattern, such as `s*`. Wildcards are permitted.

- Required? false
- Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? true

-Include <System.String[]>

Specifies, as a string array, a service or services that this cmdlet includes in the operation. The value of this parameter qualifies the Name parameter. Enter a

name element or pattern, such as `s*`. Wildcards are permitted.

None

Required?	false
Position?	named

Default value

Accept pipeline input? False

Accept wildcard characters? true

-InputObject <System.ServiceProcess.ServiceController[]>

Specifies ServiceController objects representing the services to be retrieved. Enter a variable that contains the objects, or type a command or expression that

gets the objects. You can pipe a service object to this cmdlet.

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

-Name <System.String[]>

Specifies the service names of services to be retrieved. Wildcards are permitted.

Required?	false	
Position?	0	
Default value	None	
Accept pipeline input	? True (ByPropertyName, ByValue)	
Accept wildcard characters? true		

-RequiredServices <System.Management.Automation.SwitchParameter>

Indicates that this cmdlet gets only the services that this service requires. This parameter gets the value of the ServicesDependedOn property of the service.

Required?falsePosition?namedDefault valueFalseAccept pipeline input?FalseAccept wildcard characters?true

<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.ServiceProcess.ServiceController

You can pipe a service object to this cmdlet.

System.String

You can pipe a service name to this cmdlet.

OUTPUTS

System.ServiceProcess.ServiceController

This cmdlet returns objects that represent the services on the computer.

NOTES

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

- `gsv`

This cmdlet can display services only when the current user has permission to see them. If this cmdlet does not display services, you might not have permission to

see them.

To find the service name and display name of each service on your system, type `Get-Service`. The service names appear in the Name column, and the display names

appear in the DisplayName column.

> [!NOTE] > Typically, `Get-Service` returns information about services and not driver. However, if you > specify the name of a driver, `Get-Service` returns

information about the driver. > - Enumeration doesn't include device driver services > - When a wildcard is specified, the cmdlet only returns Windows services

> - If you specify the Name or DisplayName that is an exact match to a device service name, > then the device instance is returned

When you sort in ascending order by status value, `Stopped` services appear before `Running` services. The Status property of a service is an enumerated value in

which the names of the statuses represent integer values. The sort is based on the integer value, not the name. `Running` appears before `Stopped` because

`Stopped` has a value of `1`, and `Running` has a value of `4`. For more information, see ServiceControllerStatus (/dotnet/api/system.serviceprocess.servicecontrollerstatus).

----- Example 1: Get all services on the computer ------

Get-Service

--- Example 2: Get services that begin with a search string ---

Get-Service "wmi*"

--- Example 3: Display services that include a search string ---

Get-Service -Displayname "*network*"

Example 4: Get services that begin with a search string and an exclusion

Get-Service -Name "win*" -Exclude "WinRM"

---- Example 5: Display services that are currently active ----

Get-Service | Where-Object {\$_.Status -eq "Running"}

`Get-Service` gets all the services on the computer and sends the objects down the pipeline. The `Where-Object` cmdlet, selects only the services with a Status

property that equals `Running`.

Status is only one property of service objects. To see all of the properties, type `Get-Service | Get-Member`.

------ Example 6: Get the services on a remote computer ------

Get-Service -ComputerName "Server02"

This command gets the services on the Server02 remote computer.

Because the ComputerName parameter of `Get-Service` does not use Windows PowerShell remoting, you can use this parameter even if the computer is not configured for

remoting in Windows PowerShell.

Example 7: List the services on the local computer that have dependent services

Get-Service |

Where-Object {\$_.DependentServices} |

Format-List -Property Name, DependentServices, @{

Label="NoOfDependentServices"; Expression={\$_.dependentservices.count}

}

Name : AudioEndpointBuilder

: Dhcp

DependentServices : {AudioSrv}

NoOfDependentServices : 1

Name

DependentServices : {WinHttpAutoProxySvc} NoOfDependentServices : 1

...

The `Get-Service` cmdlet gets all the services on the computer and sends the objects down the pipeline. The `Where-Object` cmdlet selects the services whose

DependentServices property isn't null.

The results are sent down the pipeline to the `Format-List` cmdlet. The Property parameter displays the name of the service, the name of the dependent services, and a

calculated property that displays the number of dependent services for each service.

----- Example 8: Sort services by property value ------

DisplayName

Get-Service "s*" | Sort-Object status

Status Name

Stopped	stisvc V	Vindows Image Acquisition (WIA)
Stopped	SwPrv	MS Software Shadow Copy Provider
Stopped	SysmonLog	Performance Logs and Alerts
Running	Spooler	Print Spooler
Running	srservice	System Restore Service
Running	SSDPSRV	SSDP Discovery Service
Running	ShellHWDetec	tion Shell Hardware Detection
Running	Schedule	Task Scheduler
Running	SCardSvr	Smart Card
Running	SamSs	Security Accounts Manager
Running	SharedAccess	Windows Firewall/Internet Connectio
Running	SENS	System Event Notification
Running	seclogon	Secondary Logon

Get-Service -Name "WinRM" -ComputerName "localhost", "Server01", "Server02" | Format-Table -Property MachineName, Status, Name, DisplayName -auto

MachineName Status Name DisplayName

----- -----

- localhost Running WinRM Windows Remote Management (WS-Management)
- Server01 Running WinRM Windows Remote Management (WS-Management)
- Server02 Running WinRM Windows Remote Management (WS-Management)

This command uses the `Get-Service` cmdlet to run a `Get-Service Winrm` command on two remote computers and the local computer (`localhost`).

The command runs on the remote computers, and the results are returned to the local computer. A pipeline operator (`|`) sends the results to the `Format-Table`

cmdlet, which formats the services as a table. The `Format-Table` command uses the Property parameter to specify the properties displayed in the table, including the

MachineName property.

----- Example 10: Get the dependent services of a service -----

Get-Service "WinRM" -RequiredServices

--- Example 11: Get a service through the pipeline operator ---

"WinRM" | Get-Service

RELATED LINKS

Online

Version:

https://learn.microsoft.com/powershell/module/microsoft.powershell.management/get-service?view=powershell-5.1&WT.mc

New-Service

Restart-Service

Resume-Service

Set-Service

Start-Service

Stop-Service

Suspend-Service