Get-Service

Module: Microsoft.PowerShell.Management

Gets the services on a local or remote computer.

In this article

PowerShell Copy

```
Get-Service
   [[-Name] <String[]>]
   [-ComputerName <String[]>]
   [-DependentServices]
   [-RequiredServices]
   [-Include <String[]>]
   [-Exclude <String[]>]
   [<CommonParameters>]
```

PowerShell Copy

```
Get-Service
   [-ComputerName <String[]>]
   [-DependentServices]
   [-RequiredServices]
   -DisplayName <String[]>
   [-Include <String[]>]
   [-Exclude <String[]>]
   [<CommonParameters>]
```

PowerShell Copy

```
Get-Service
   [-ComputerName <String[]>]
   [-DependentServices]
   [-RequiredServices]
   [-Include <String[]>]
   [-Exclude <String[]>]
   [-InputObject <ServiceController[]>]
   [<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.

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

Examples

Example 1: Get all services on the computer

PowerShell Copy

```
PS C:\> Get-Service
```

This command gets all of the services on the computer. It behaves as though you typed `Get-Service *`. The default display shows the status, service name, and display name of each service.

Example 2: Get services that begin with a search string

PowerShell Copy

```
PS C:\> Get-Service "wmi*"
```

This command retrieves services with service names that begin with WMI (the acronym for Windows Management Instrumentation).

Example 3: Display services that include a search string

PowerShell Copy

```
PS C:\> Get-Service -Displayname "*network*"
```

This command displays services with a display name that includes the word network. Searching the display name finds network-related services even when the service name does not include "Net", such as xmlprov, the Network Provisioning Service.

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

PowerShell Copy

```
PS C:\> Get-Service -Name "win*" -Exclude "WinRM"
```

These commands get only the services with service names that begin with win, except for the WinRM service.

Example 5: Display services that are currently active

PowerShell Copy

```
PS C:\> Get-Service | Where-Object {$_.Status -eq "Running"}
```

This command displays only the services that are currently active. It uses the **Get-Service** cmdlet to get all of the services on the computer. The pipeline operator (|) passes the results to the Where-Object cmdlet, which 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

PowerShell Copy

```
PS C:\> 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

PowerShell Copy

```
PS C:\> Get-Service | Where-Object {$_.DependentServices} | Format-List -Property Name,
DependentServices, @{Label="NoOfDependentServices"; Expression=
{$_.dependentservices.count}}

Name                  : AudioEndpointBuilder
DependentServices     : {AudioSrv}
NoOfDependentServices : 1
```

```
Name                  : Dhcp
DependentServices     : {WinHttpAutoProxySvc}
NoOfDependentServices : 1
...
```

This example lists the services on the computer that have dependent services.

The first command uses the **Get-Service** cmdlet to get the services on the computer. A pipeline operator (|) sends the services to the **Where-Object** cmdlet, which selects the services whose **DependentServices** property is not null.

Another pipeline operator sends the results to the Format-List cmdlet. The command uses its *Property* parameter to display the name of the service, the name of the dependent services, and a calculated property that displays the number of dependent services that each service has.

Example 8: Sort services by property value

PowerShell Copy

```
PS C:\> Get-Service "s*" | Sort-Object status

Status   Name               DisplayName
------   ----               -----------
Stopped  stisvc             Windows 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  ShellHWDetection   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

PS C:\> Get-Service "s*" | Sort-Object status -Descending

Status   Name               DisplayName
------   ----               -----------
Running  ShellHWDetection   Shell Hardware Detection
Running  SharedAccess       Windows Firewall/Internet Connectio...
Running  Spooler            Print Spooler
Running  SSDPSRV            SSDP Discovery Service
Running  srservice          System Restore Service
Running  SCardSvr           Smart Card
Running  SamSs              Security Accounts Manager
Running  Schedule           Task Scheduler
Running  SENS               System Event Notification
Running  seclogon           Secondary Logon
Stopped  SysmonLog          Performance Logs and Alerts
```

```
Stopped  SwPrv               MS Software Shadow Copy Provider
Stopped  stisvc             Windows Image Acquisition (WIA)
```

This command shows that when you sort services in ascending order by the value of their **Status** property, stopped services appear before running services. This happens because the value of Status is an enumeration, in which Stopped has a value of 1, and Running has a value of 4.

To list running services first, use the *Descending* parameter of the Sort-Object cmdlet.

Example 9: Get services on multiple computers

PowerShell Copy

```
PS C:\> 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

PowerShell Copy

```
PS C:\> Get-Service "WinRM" -RequiredServices
```

This command gets the services that the WinRM service requires.

The command returns the value of the **ServicesDependedOn** property of the service.

Example 11: Get a service through the pipeline operator

PowerShell Copy

```
PS C:\> "WinRM" | Get-Service
```

This command gets the WinRM service on the local computer. This example shows that you can pipe a service name string (enclosed in quotation marks) to **Get-Service**.

Required Parameters

`-DisplayName`

Specifies, as a string array, the display names of services to be retrieved. Wildcards are permitted. By default, this cmdlet gets all services on the computer.

Type:
String[]

Position:
Named

Default value:
None

Accept pipeline input:
False

Accept wildcard characters:
False

Optional Parameters

`-ComputerName`

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.

Type:
String[]

Aliases:
Cn

Position:
Named

Default value:
None

Accept pipeline input:
True (ByPropertyName)

Accept wildcard characters:
False

-DependentServices

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

By default, this cmdlet gets all services.

Type:
SwitchParameter

Aliases:
DS

Position:
Named

Default value:
None

Accept pipeline input:
False

Accept wildcard characters:
False

-Exclude

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.

Type:
String[]

Position:
Named

Default value:
None

Accept pipeline input:
False

Accept wildcard characters:
False

-Include

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.

Type:
String[]

Position:
Named

Default value:
None

Accept pipeline input:
False

Accept wildcard characters:
False

-InputObject

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 also pipe a service object to this cmdlet.

Type:
ServiceController[]

Position:
Named

Default value:
None

Accept pipeline input:
True (ByValue)

Accept wildcard characters:
False

-Name

Specifies the service names of services to be retrieved. Wildcards are permitted. By default, this cmdlet gets all of the services on the computer.

Type:
String[]

Aliases:
ServiceName

Position:
0

Default value:
None

Accept pipeline input:
True (ByPropertyName, ByValue)

Accept wildcard characters:
False

-RequiredServices

Indicates that this cmdlet gets only the services that this service requires.

This parameter gets the value of the **ServicesDependedOn** property of the service. By default, this cmdlet gets all services.

Type:
SwitchParameter

Aliases:
SDO, ServicesDependedOn

Position:
Named

Default value:
None

Accept pipeline input:
False

Accept wildcard characters:
False

Inputs

System.ServiceProcess.ServiceController, System.String

You can pipe a service object or a service name to this cmdlet.

Outputs

System.ServiceProcess.ServiceController

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

Notes

- You can also refer to **Get-Service** by its built-in alias, "gsv". For more information, see about_Aliases.

  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.

  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".
-

Related Links

- New-Service
- Restart-Service
- Resume-Service
- Set-Service
- Start-Service
- Stop-Service
- Suspend-Service