



python



PowerShell

FPDF Library  
PDF generator

*Full credit is given to the above companies including the OS that this PDF file was generated!*

### ***PowerShell Get-Help on command 'Get-ScheduledJobOption'***

***PS C:\Users\wahid> Get-Help Get-ScheduledJobOption***

#### NAME

Get-ScheduledJobOption

#### SYNOPSIS

Gets the job options of scheduled jobs.

#### SYNTAX

Get-ScheduledJobOption [-Id] <System.Int32> [<CommonParameters>]

Get-ScheduledJobOption [-InputObject]

<Microsoft.PowerShell.ScheduledJob.ScheduledJobDefinition> [<CommonParameters>]

Get-ScheduledJobOption [-Name] <System.String> [<CommonParameters>]

#### DESCRIPTION

The `Get-ScheduledJobOption` cmdlet gets the job options of scheduled jobs.

You can use this command to examine the job options or to pipe the job options to other cmdlets.

Job options are not saved to disk independently; they are part of a scheduled job. To get the job options of a scheduled job, specify the scheduled job.

Use the parameters of the ``Get-ScheduledJobOption`` cmdlet to identify the scheduled job. You can identify scheduled jobs by their names or identification numbers, or by entering or piping `ScheduledJob` objects, such as those that are returned by the ``Get-ScheduledJob`` cmdlet, to ``Get-ScheduledJobOption``.

``Get-ScheduledJobOption`` is one of a collection of job scheduling cmdlets in the `PSScheduledJob` module that is included in Windows PowerShell.

For more information about Scheduled Jobs, see the About topics in the `PSScheduledJob` module. Import the `PSScheduledJob` module and then type: ``Get-Help about_Scheduled*`` or see `about_Scheduled_Jobs` (`About/about_Scheduled_Jobs.md`).

This cmdlet was introduced in Windows PowerShell 3.0.

## PARAMETERS

`-Id <System.Int32>`

Specifies the identification number of a scheduled job.

``Get-ScheduledJobOption`` gets the job options of the specified scheduled job.

To get the identification numbers of scheduled jobs on the local computer or a remote computer, use the ``Get-ScheduledJob`` cmdlet.

`-InputObject <Microsoft.PowerShell.ScheduledJob.ScheduledJobDefinition>`

Specifies a scheduled job. Enter a variable that contains a `ScheduledJob` object or type a command or expression that gets a `ScheduledJob` object, such as a ``Get-ScheduledJob`` command. You can also pipe a `ScheduledJob`

object to `Get-ScheduledJobOption`.

`-Name <System.String>`

Specifies the names of scheduled jobs. `Get-ScheduledJobOption` gets the job options of the specified scheduled job. Wildcards are supported.

To get the names of scheduled jobs on the local computer or a remote computer, use the `Get-ScheduledJob` cmdlet.

`<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>).

----- Example 1: Get job options -----

```
Get-ScheduledJobOption -Name "**Backup**"
```

----- Example 2: Get all job options -----

```
Get-ScheduledJob | Get-ScheduledJobOption
```

The example uses the `Get-ScheduledJob` cmdlet to get the scheduled jobs on the local computer. A pipeline operator ( `|` ) sends the scheduled jobs to the `Get-ScheduledJobOption` cmdlet, which gets the job options of each scheduled job.

----- Example 3: Get selected job options -----

```
Get-ScheduledJob | Get-ScheduledJobOption | Where {$_.RunElevated -and !$_.WaketoRun}
```

```
Get-ScheduledJob | Get-ScheduledJobOption | Where {$_.RunElevated -and  
!$_.WaketoRun} |  
    ForEach-Object {$_.JobDefinition}
```

The first command gets job options in which the RunElevated property has a value of \$True and the RunWithoutNetwork property has a value of `false`. The output shows the JobOptions object that was selected.

The second command shows how to find to which scheduled job the job options belong. This command uses a pipeline operator (|) to send the selected job options to the `ForEach-Object` cmdlet, which gets the JobDefinition property of each options object. The JobDefinition property contains the originating job object.

----- Example 4: Use job options to create a new job -----

```
$Opts = Get-ScheduledJobOption -Name "BackupTestLogs"  
Register-ScheduledJob -Name "Archive-Scripts" -FilePath  
"\\Srv01\Scripts\ArchiveScripts.ps1" -ScheduledJobOption $Opts
```

The first command uses `Get-ScheduledJobOption` to get the jobs options of the BackupTestLogs scheduled job. The command saves the options in the `\$Opts` variable.

The second command uses `Register-ScheduledJob` cmdlet to create a new scheduled job. The value of the ScheduledJobOption parameter is the options object in the `\$Opts` variable.

----- Example 5: Get job options from a remote computer -----

```
$O = Invoke-Command -ComputerName "Srv01" -ScriptBlock {Get-ScheduledJob -Name  
"DataDemon" }
```

This command uses the `Invoke-Command` cmdlet to get the scheduled job options of the DataDemon job on the Srv01 computer. The command saves the options in

the `\$O` variable.

## REMARKS

To see the examples, type: "get-help Get-ScheduledJobOption -examples".

For more information, type: "get-help Get-ScheduledJobOption -detailed".

For technical information, type: "get-help Get-ScheduledJobOption -full".

For online help, type: "get-help Get-ScheduledJobOption -online"