



### ***Windows PowerShell Get-Help on Cmdlet 'Set-ScheduledTask'***

***PS:\>Get-HELP Set-ScheduledTask -Full***

#### **NAME**

Set-ScheduledTask

#### **SYNOPSIS**

Modifies a scheduled task.

#### **SYNTAX**

```
Set-ScheduledTask [-TaskName] <String> [[-TaskPath] <String>] [[-Action] <CimInstance[]>] [[-Trigger] <CimInstance[]>]
[[[-Settings] <CimInstance>] [[-User] <String>]
[[[-Password] <String>] [-AsJob] [-CimSession <CimSession[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

Set-ScheduledTask [-TaskName] <String> [[-TaskPath] <String>] [[-Action] <CimInstance[]>] [[-Trigger] <CimInstance[]>]
[[[-Settings] <CimInstance>] [[-Principal]
<CimInstance>] [-AsJob] [-CimSession <CimSession[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

Set-ScheduledTask [-InputObject] <CimInstance> [[-User] <String>] [[-Password] <String>] [-AsJob] [-CimSession
<CimSession[]>] [-ThrottleLimit <Int32>]
[<CommonParameters>]
```

## DESCRIPTION

The Set-ScheduledTask cmdlet modifies a task definition.

You can make changes to a task definition even if an instance of the task is running. The changes do not affect the current instance.

## PARAMETERS

**-Action < CimInstance[] >**

Specifies an array of work items to be performed by the task. If you specify several actions, the computer runs them in order. You can specify up to 32 actions.

Required?	false
Position?	2
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

**-AsJob [< SwitchParameter >]**

Runs the cmdlet as a background job. Use this parameter to run commands that take a long time to complete.

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

**-CimSession < CimSession[] >**

Runs the cmdlet in a remote session or on a remote computer. Enter a computer name or a session object, such as the output of a New-CimSession

(<https://go.microsoft.com/fwlink/p/?LinkId=227967>) or

[Get-CimSession](<https://go.microsoft.com/fwlink/p/?LinkId=227966>)cmdlet. The default is the current session *Page 2/8*

on the local computer.

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

#### -InputObject <CimInstance>

Specifies the input object that is used in a pipeline command.

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

#### -Password <String>

Specifies a password for the <run as> user. The password is ignored for the well-known system accounts.

Well-known accounts are: NT AUTHORITY\SYSTEM, NT AUTHORITY\LOCALSERVICE, NT AUTHORITY\NETWORKSERVICE, and the well-known security identifiers (SIDs) for all three accounts.

Required?	false
Position?	6
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

#### -Principal <CimInstance>

Specifies the security context in which a task is run.

Required?	false
Position?	5
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-Settings <CimInstance>

Specifies a configuration object that the Task Scheduler service uses to determine how to run a task.

Required?	false
Position?	4
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-TaskName <String>

Specifies the name of a scheduled task.

Required?	true
Position?	0
Default value	None
Accept pipeline input?	False
Accept wildcard characters?	false

-TaskPath <String>

Specifies an array of one or more paths for scheduled tasks in Task Scheduler namespace. You can use "\*" for a wildcard character query. You can use \ for the root folder. To specify a full TaskPath you need to include the leading and trailing \ \*. If you do not specify a path, the cmdlet uses the root folder.

Required?	false
Position?	1
Default value	None

Accept pipeline input? False

Accept wildcard characters? false

#### -ThrottleLimit <Int32>

Specifies the maximum number of concurrent operations that can be established to run the cmdlet. If this parameter is omitted or a value of `0` is entered, then

Windows PowerShell calculates an optimum throttle limit for the cmdlet based on the number of CIM cmdlets that are running on the computer. The throttle limit

applies only to the current cmdlet, not to the session or to the computer.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

#### -Trigger <CimInstance[]>

Specifies an array of one or more trigger objects that cause a scheduled task to start.

A trigger is a set of criteria that, when met, starts a scheduled task. You can use a time-based trigger or an event-based trigger to start a task and a task can

be started by one or more triggers. A task can have up to 48 triggers. For more information about triggers, see Triggers

(<https://technet.microsoft.com/en-us/library/cc748841.aspx>).

Required? false

Position? 3

Default value None

Accept pipeline input? False

Accept wildcard characters? false

#### -User <String>

Specifies the name of a <run as> user account to use when you run the task.

Required? false  
Position? 5  
Default value None  
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

OUTPUTS

Microsoft.Management.Infrastructure.CimInstance#MSFT\_ScheduledTask

NOTES

----- Example 1: Modify a trigger in a scheduled task -----

```
PS C:\> $Time = New-ScheduledTaskTrigger -At 12:00 -Once  
PS C:\> Set-ScheduledTask -TaskName "SoftwareScan" -Trigger $Time
```

TaskPath	TaskName
-----	-----
\	SoftwareScan

In this example, the first command uses the `New-ScheduledTaskTrigger` cmdlet to define a time trigger, to which the `$Time` variable is assigned.

The second command adds (or replaces) the `$Time` trigger in the scheduled task `SoftwareScan`.

-- Example 2: Modify settings in a scheduled task definition --

```
PS C:\> $Act1 = New-ScheduledTaskAction -Execute "Notepad.exe"
```

```
PS C:\> $Act2 = New-ScheduledTaskAction -Execute "Calc.exe"
```

```
PS C:\> Set-ScheduledTask "DeployTools" -Action $Act1,$Act2
```

TaskPath	TaskName
----------	----------

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

\	DeployTools
---	-------------

In this example, the set of commands uses cmdlets and variables to modify a scheduled task.

The first command uses the `New-ScheduledTaskAction` cmdlet to define an action, to which the `$Act1` variable is assigned.

The second command uses the `New-ScheduledTaskAction` cmdlet to define a second action, to which the `$Act2` variable is assigned.

The third command adds the two actions to the scheduled task `DeployTools`.

## RELATED LINKS

Online

Version:

[https://learn.microsoft.com/powershell/module/scheduledtasks/set-scheduledtask?view=windowsserver2022-ps&wt.mc\\_id=ps-gethelp](https://learn.microsoft.com/powershell/module/scheduledtasks/set-scheduledtask?view=windowsserver2022-ps&wt.mc_id=ps-gethelp)

[Disable-ScheduledTask](#)

[Enable-ScheduledTask](#)

[Export-ScheduledTask](#)

[Get-ScheduledTask](#)

[New-ScheduledTask](#)

[New-ScheduledTaskAction](#)

New-ScheduledTaskTrigger

Register-ScheduledTask

Start-ScheduledTask

Stop-ScheduledTask

Unregister-ScheduledTask