



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 'New-ScheduledTask'

PS C:\Users\wahid> Get-Help New-ScheduledTask

NAME

New-ScheduledTask

SYNOPSIS

Creates a scheduled task instance.

SYNTAX

```
New-ScheduledTask [[-Action] < CimInstance[]>] [[-Trigger] < CimInstance[]>]
[[ -Settings] < CimInstance>] [[-Principal] < CimInstance>] [[-Description]
<String>] [-AsJob] [-CimSession < CimSession[]>] [-ThrottleLimit < Int32>]
[<CommonParameters>]
```

DESCRIPTION

The New-ScheduledTask cmdlet creates an object that contains the definition of a scheduled task. New-ScheduledTask does not automatically register the object with the Task Scheduler service.

You can register a task to run any of the following application or file types:

Win32 applications, Win16 applications, OS/2 applications, MS-DOS

applications, batch files (.bat), command files (.cmd), or any properly registered file type.

PARAMETERS

-Action <CimInstance[]>

Specifies an array of work items for a task to run. When you specify multiple actions, they run sequentially. A task can have up to 32 actions.

-AsJob [<SwitchParameter>]

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

-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 (/powershell/module/cimcmdlets/new-cimsession) or

[Get-CimSession](<https://go.microsoft.com/fwlink/p/?LinkId=227966>)cmdlet.

The default is the current session on the local computer.

-Description <String>

Briefly describes the task.

-Principal <CimInstance>

Specifies the security context in which a task runs.

-Settings <CimInstance>

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

-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.

-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 starts a scheduled task when the criteria are met. You can use a time-based trigger or an event-based trigger to start a task, and one or more triggers can start a task. A task can have up to 48 triggers. For more information about triggers, see [Triggers \(https://technet.microsoft.com/library/cc748841.aspx\)](https://technet.microsoft.com/library/cc748841.aspx).

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable. For more information, see [about_CommonParameters \(https://go.microsoft.com/fwlink/?LinkID=113216\)](https://go.microsoft.com/fwlink/?LinkID=113216).

Example 1: Define a scheduled task and register it at a later time

```
PS C:\> $action = New-ScheduledTaskAction -Execute "Taskmgr.exe"
PS C:\> $trigger = New-ScheduledTaskTrigger -AtLogon
PS C:\> $principal = "Contoso\Administrator"
PS C:\> $settings = New-ScheduledTaskSettingsSet
PS C:\> $task = New-ScheduledTask -Action $action -Principal $principal
-Trigger $trigger -Settings $settings
PS C:\> Register-ScheduledTask T1 -InputObject $task
```

In this example, the set of commands uses several cmdlets and variables to define and then register a scheduled task.

The first command uses the `New-ScheduledTaskAction` cmdlet to assign the executable file ``tskmgr.exe`` to the variable ``$action``.

The second command uses the `New-ScheduledTaskTrigger` cmdlet to assign the value ``AtLogon`` to the variable ``$trigger``.

The third command assigns the principal of the scheduled task ``Contoso\Administrator`` to the variable ``$principal``.

The fourth command uses the `New-ScheduledTaskSettingsSet` cmdlet to assign a task settings object to the variable ``$settings``.

The fifth command creates a new task and assigns the task definition to the variable ``$task``.

The sixth command (hypothetically) runs at a later time. It registers the new scheduled task and defines it by using the ``$task`` variable.

--- Example 2: Define a scheduled task with multiple actions ---

```
PS C:\> $actions = (New-ScheduledTaskAction -Execute 'foo.ps1'),  
(New-ScheduledTaskAction -Execute 'bar.ps1')
```

```
PS C:\> $trigger = New-ScheduledTaskTrigger -Daily -At '9:15 AM'
```

```
PS C:\> $principal = New-ScheduledTaskPrincipal -UserId 'DOMAIN\user'  
-RunLevel Highest
```

```
PS C:\> $settings = New-ScheduledTaskSettingsSet -RunOnlyIfNetworkAvailable  
-WakeToRun
```

```
PS C:\> $task = New-ScheduledTask -Action $actions -Principal $principal  
-Trigger $trigger -Settings $settings
```

```
PS C:\> Register-ScheduledTask 'baz' -InputObject $task
```

This example creates and registers a scheduled task that runs two PowerShell

scripts daily at 09:15 AM.

REMARKS

To see the examples, type: "get-help New-ScheduledTask -examples".

For more information, type: "get-help New-ScheduledTask -detailed".

For technical information, type: "get-help New-ScheduledTask -full".

For online help, type: "get-help New-ScheduledTask -online"