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

PowerShell Get-Help on command 'Remove-JobTrigger'

PS C:\Users\wahid> Get-Help Remove-JobTrigger

NAME

Remove-JobTrigger

SYNOPSIS

Delete job triggers from scheduled jobs.

SYNTAX

Remove-JobTrigger [-Id] <System.Int32[]> [-TriggerId <System.Int32[]>]
[<CommonParameters>]

Remove-JobTrigger [-InputObject]
<Microsoft.PowerShell.ScheduledJob.ScheduledJobDefinition[]> [-TriggerId
<System.Int32[]>] [<CommonParameters>]

Remove-JobTrigger [-Name] <System.String[]> [-TriggerId <System.Int32[]>]
[<CommonParameters>]

DESCRIPTION

The `Remove-JobTrigger` cmdlet deletes job triggers from scheduled jobs.

A job trigger defines a recurring schedule or conditions for starting a scheduled job. To manage job triggers, use the New-JobTrigger, Add-JobTrigger, Set-JobTrigger, and `Set-ScheduledJob` cmdlets.

Use the Name , ID , or InputObject parameters of `Remove-JobTrigger` to identify the scheduled jobs from which the triggers are removed. Use the TriggerID parameter to identify the job triggers to delete. By default, `Remove-JobTrigger` deletes all job triggers of a scheduled job.

`Remove-JobTrigger` 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 numbers of the scheduled jobs. `Remove-JobTrigger` deletes job triggers from the specified scheduled jobs.

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

-InputObject <Microsoft.PowerShell.ScheduledJob.ScheduledJobDefinition[]>

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

objects to `Remove-JobTrigger`.

-Name <System.String[]>

Specifies the names of the scheduled jobs. `Remove-JobTrigger` deletes the job triggers from the specified scheduled jobs. Wildcards are supported.

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

-TriggerId <System.Int32[]>

Deletes only the specified job triggers. By default, `Remove-JobTrigger` deletes all triggers from the scheduled jobs. Use this parameter when the scheduled jobs have multiple job triggers.

Enter the trigger IDs of one or more job triggers of a scheduled job. If you specify multiple scheduled jobs, `Remove-JobTrigger` deletes the job trigger with the specified ID from all scheduled jobs.

<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: Delete all job triggers ----------

Remove-JobTrigger -Name "Test*"

This command deletes all job triggers from scheduled job that have names that begin with Test.

---------- Example 2: Delete selected job triggers ----------

Remove-JobTrigger -Name "BackupArchive" -TriggerID 3

This command deletes only the third trigger (ID = 3) from the BackupArchive scheduled job.

Example 3: Delete AtStartup job triggers from all scheduled jobs

```
function Delete-AtStartup
{
    Get-ScheduledJob | Get-JobTrigger | Where-Object {$_.Frequency -eq "AtStartup"} | ForEach-Object { Remove-JobTrigger -InputObject $_.JobDefinition -TriggerID $_.ID}
}
```

This function deletes all AtStartup job triggers from all jobs on the local computer. To use the function, run the function in your session and then type `Delete-AtStartup`.

The `Delete-AtStartup` function contains a single command. The command uses the `Get-ScheduledJob` cmdlet to get the scheduled jobs on the local computer. A pipeline operator (`|`) sends the scheduled jobs to the `Get-JobTrigger` cmdlet, which gets all of the job triggers from each of the scheduled jobs. A pipeline operator sends the job triggers to the `Where-Object` cmdlet, which selects job triggers where the value of the Frequency property of the job trigger equals AtStartup. JobTrigger objects have a JobDefinition property that contains the scheduled job that they trigger. The remainder of the command uses that valuable feature.

A pipeline operator sends the AtStartup job triggers to the `ForEach-Object` cmdlet, which runs a `Remove-JobTrigger` command on each AtStartup trigger. The value of the InputObject parameter of `Remove-JobTrigger` is the scheduled job in the JobDefinition property of the job trigger. The value of the TriggerID parameter is the identifier in the ID property of the job trigger.

- Example 4: Delete a job trigger from a remote scheduled job -

Invoke-Command -ComputerName "Server01" { Remove-JobTrigger -ID 38 -TriggerID 1 }

This command deletes the first job trigger from the Inventory job on the Server01 computer.

The command uses the `Invoke-Command` cmdlet to run the `Remove-JobTrigger` cmdlet on the Server01 computer. The `Remove-JobTrigger` cmdlet uses the ID parameter to identify the Inventory scheduled job and the TriggerID parameter to specify the first trigger. The ID parameter is especially useful when multiple scheduled jobs have the same or similar names.

REMARKS

To see the examples, type: "get-help Remove-JobTrigger -examples".
For more information, type: "get-help Remove-JobTrigger -detailed".
For technical information, type: "get-help Remove-JobTrigger -full".
For online help, type: "get-help Remove-JobTrigger -online"