Windows PowerShell Get-Help on Cmdlet 'Get-IscsiSession'

PS:\>Get-HELP Get-IscsiSession -Full

NAME

Get-IscsiSession

SYNOPSIS

Retrieves information about established iSCSI sessions.

SYNTAX

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-NumberOfConnections <UInt32[]>] [-SessionIdentifier <String[]>] [-TargetSideIdentifier <String[]>]

[-ThrottleLimit <Int32>] [<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-InitiatorSideIdentifier <String[]>] [-NumberOfConnections <UInt32[]>] [-ThrottleLimit <Int32>]

[<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-IscsiTarget <CimInstance>] [-NumberOfConnections <UInt32[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-InitiatorPort <CimInstance>] [-NumberOfConnections

<UInt32[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-IscsiTargetPortal <CimInstance>] [-NumberOfConnections <UInt32[]>] [-ThrottleLimit <Int32>]
[<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-Disk <CimInstance>] [-NumberOfConnections <UInt32[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

Get-IscsiSession [-AsJob] [-CimSession <CimSession[]>] [-IscsiConnection <CimInstance>] [-NumberOfConnections <UInt32[]>] [-ThrottleLimit <Int32>] [<CommonParameters>]

DESCRIPTION

The Get-IscsiSession cmdlet returns information about iSCSI sessions.

There is an association between the iSCSI session and the disk object. It is possible to return all disks connected by using a specific iSCSI session by running the
following command:

PS C:\> `Get-iSCSISession | Get-Disk`

PARAMETERS

-AsJob [<SwitchParameter>]

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

The cmdlet immediately returns an object that represents the job and then displays the command prompt. You can continue to work in the session while the job
completes. To manage the job, use the `*-Job` cmdlets. To get the job results, use the Receive-Job (https://go.microsoft.com/fwlink/?LinkID=113372)cmdlet.

For more information about Windows PowerShell background jobs, see about_Jobs

(https://go.microsoft.com/fwlink/?LinkID=113251).

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 on the local computer.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Disk <CimInstance>

Accepts a MSFT disk object as an input. The MSFT disk object is output from the Get-Disk cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-InitiatorPort <CimInstance>

Accepts a MSFT initiator port object as an input. The MSFT initiator port object is output from the Get-InitiatorPort cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-InitiatorSideIdentifier <String[]>

Specifies the initiator side identifier.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-IscsiConnection <CimInstance>

Accepts a MSFT iSCSI connection object as an input. The MSFT iSCSI connection object is output from the Get-IscsiConnection cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-IscsiTarget <CimInstance>

Accepts a MSFT iSCSI target object as an input. The MSFT iSCSI target object is output from the Get-IscsiTarget cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-IscsiTargetPortal <CimInstance>

Accepts a MSFT iSCSI target portal object as an input. The MSFT iSCSI target portal object is output from the Get-IscsiTargetPortal cmdlet.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByValue)

Accept wildcard characters? false

-NumberOfConnections <UInt32[]>

Specifies the number of connections.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-SessionIdentifier <String[]>

Specifies the session identifier.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-TargetSideIdentifier <String[]>

Specifies the target side identifier.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

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 PowerShellr 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

<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

Microsoft.Management.Infrastructure.CimInstance#MSFT_DISK

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

Microsoft.Management.Infrastructure.CimInstance#MSFT_InitiatorPort

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

Microsoft.Management.Infrastructure.CimInstance#MSFT_IscsiConnection

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

Microsoft.Management.Infrastructure.CimInstance#MSFT_IscsiTarget

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

Microsoft.Management.Infrastructure.CimInstance#MSFT_IscsiTargetPortal

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

OUTPUTS

Microsoft.Management.Infrastructure.CimInstance#MSFT_IscsiSession

The `Microsoft.Management.Infrastructure.CimInstance` object is a wrapper class that displays Windows Management Instrumentation (WMI) objects. The path after the

pound sign (`#`) provides the namespace and class name for the underlying WMI object.

NOTES

------- Example 1: Get information about iSCSI sessions -------

```
PS C:\> Get-IscsiSession

AuthenticationType      : NONE
InitiatorInstanceName   : ROOT\ISCSIPRT\0000_0
InitiatorNodeAddress    : iqn.1991-05.com.microsoft:deepcore.contoso.com
InitiatorPortalAddress  :
InitiatorSideIdentifier : 400001370001
IsConnected             : True
IsDataDigest            : False
IsDiscovered            : True
IsHeaderDigest          : False
IsMultipathEnabled      : False
IsPersistent            : True
NumberOfConnections     : 1
SessionIdentifier       : fffffa800d008430-4000013700000001
TargetNodeAddress       : iqn.1991-05.com.contoso:testiscsi-deepcore-target
TargetSideIdentifier    : 0200

AuthenticationType      : NONE
InitiatorInstanceName   : ROOT\ISCSIPRT\0000_0
InitiatorNodeAddress    : iqn.1991-05.com.contoso:deepcore.contoso.com
InitiatorPortalAddress  :
InitiatorSideIdentifier : 400001370004
IsConnected             : True
IsDataDigest            : False
IsDiscovered            : True
IsHeaderDigest          : False
IsMultipathEnabled      : False
IsPersistent            : True
NumberOfConnections     : 1
SessionIdentifier       : fffffa800d008430-4000013700000002
TargetNodeAddress       : iqn.1991-05.com.contoso:testiscsi-deepcore-target
TargetSideIdentifier    : 0100
```

AuthenticationType : NONE

InitiatorInstanceName : ROOT\ISCSIPRT\0000_0

InitiatorNodeAddress : iqn.1991-05.com.contoso:deepcore.contoso.com

InitiatorPortalAddress :

InitiatorSideIdentifier : 400001370002

IsConnected : True

IsDataDigest : False

IsDiscovered : True

IsHeaderDigest : False

IsMultipathEnabled : False

IsPersistent : True

NumberOfConnections : 1

SessionIdentifier : fffffa800d008430-4000013700000003

TargetNodeAddress : iqn.1991-05.com.contoso:testiscsi-deepcore-target

TargetSideIdentifier : 0300

This command returns information about iSCSI sessions.

RELATED LINKS

Online Version:

https://learn.microsoft.com/powershell/module/iscsi/get-iscsisession?view=windowsserver2022-ps&wt.mc_id=ps-gethelp

iSCSI on TechNet

Storage on TechNet https://go.microsoft.com/fwlink/?linkid=191356

Get-IscsiConnection

Get-IscsiTarget

Get-IscsiTargetPortal