



# REDUXIO BEST PRACTICES

FOR WINDOWS SERVERS

For more information, refer to Reduxio Systems Inc. website at <http://www.reduxio.com>.

If you have comments about this documentation, submit your feedback to [docs@reduxio.com](mailto:docs@reduxio.com).

© 2017 Reduxio Systems Inc. All rights reserved. No portions of this document may be reproduced without prior written consent of Reduxio Systems Inc..

Reduxio® , the Reduxio logo, NoDup® , BackDating™ , Tier-X™ , StorSense™ , NoRestore™ , TimeOS™ , and NoMigrate™ are trademarks or registered trademarks of Reduxio in the United States and/or other countries.

Linux is a registered trademark of Linus Torvalds.

Windows is a registered trademark of Microsoft Corporation.

UNIX is a registered trademark of The Open Group.

ESX and VMWare are registered trademarks of VMWare, Inc.

Amazon and Amazon Web Services and AWS are trademarks of Amazon Services LLC and/or its affiliates.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

The Reduxio system hardware, software, user interface and/or information contained herein are Reduxio Systems Inc. proprietary and confidential. Any and all rights including all intellectual property rights associated therewith are reserved and shall remain with Reduxio Systems Inc. Rights to use, if any, shall be subject to the acceptance of the End User License Agreement provided with the system.

Information in this document is subject to change without notice.

Reduxio Systems Inc.

111 Pine Avenue

South San Francisco CA, 94080

United States

[www.reduxio.com](http://www.reduxio.com)

# Revisions

---

---

**July 8, 2016** Initial revision.

---

**July 27, 2016** Added manual configuration.

---

**Feb 27, 2017** Fixed PDF properties.

---

**Nov 2, 2017** Fixed typos in PowerShell commands.

---

# Table of Contents

---

<b>Overview</b>	5
ALUA Support	5
Supported Windows Editions	5
<b>Configuration</b>	6
Reduxio StorKit for Microsoft Windows Server (RSMS)	6
Manual Configuration	6
Device Claiming	7
Load Balancing Policy	7
Required Registry Settings	7
<b>Validation</b>	10
Validating Windows Registry Settings	10
Validating Microsoft Multipath I/O	11
<b>Manual Multipath Configuration</b>	14
Identify Reduxio iSCSI IP Addresses	14
Configure iSCSI Initiator	14
Configure Multipathing	19
Validating Multipathing Connection	26
<b>Conclusions</b>	28
Reduxio Documentation	28
Microsoft Documentation	28

# Overview

---

Reduxio® is a Microsoft Storage Partner, and Reduxio TimeOS™ is certified for Windows Server 2012. Reduxio HX Series storage systems are fully integrated with core Windows features: the Microsoft iSCSI Software Initiator and Microsoft MPIO (Multipath I/O). The settings required for proper operations are described in this document.

## ALUA Support

Reduxio HX Series storage supports the Asymmetric Logical Unit Access (ALUA) standard, providing native path failover and load balancing. Microsoft DSM, provided with the Windows operating system, supports ALUA. No additional component is required. However, minimal configuration must be done to enable high-availability and load balancing for Reduxio storage.

## Supported Windows Editions

The server class Windows operating systems include ALUA support and are supported:

- Windows Server 2008 R2
- Windows Server 2012, 2012 R2
- Windows Server 2016

The following client class operating systems do not include ALUA support and are not supported:

- Windows 7, 8, 8.1
- Windows 10



Examples in this document are taken from Windows Server 2012 R2. Users of other supported versions, such as Windows Server 2016, may notice some differences in appearance or nomenclature.

---

# Configuration

---

This section includes the following topics:

---

<b>Reduxio StorKit for Microsoft Windows Server (RSMS)</b>	6
<b>Manual Configuration</b>	6
<b>Device Claiming</b>	7
<b>Load Balancing Policy</b>	7
<b>Required Registry Settings</b>	7

## Reduxio StorKit for Microsoft Windows Server (RSMS)

To simplify the configuration of Microsoft Windows Server and the Reduxio system, the Reduxio StorKit for Microsoft Windows Server includes `ReduxioWindows_HAT`. This PowerShell-based tool configures the initial settings required for proper iSCSI operations with Reduxio HX Series storage, and for connection to a Reduxio HX Series system.



The steps and best practices provided in the following sections are automatically performed by `ReduxioWindows_HAT`. They are described in this document for reference only.

---

## Manual Configuration

To manually configure iSCSI on Windows Server with the necessary settings, install both the iSCSI Initiator and Microsoft Multipath I/O (MPIO).

To start the iSCSI Initiator:

- 
- |    |   |
|----|---|
| 1. | Launch PowerShell.                          |
| 2. | Type <code>StartiSCSI</code> at the prompt: |

```
PS C:\> StartiSCSI
```

To install MPIO:

1. Launch PowerShell.
2. Type `Install-WindowsFeature -Name Multipath-IO` at the prompt:

```
PS C:\> Install-WindowsFeature -Name Multipath-IO
```

## Device Claiming

By default, iSCSI devices are not claimed by the Multipath I/O code.

To enable automatic claiming of Reduxio devices for MPIO using the command prompt type `mpclaim -n -i -d "REDUXIO TCAS"` at the PowerShell prompt.

```
PS C:\> mpclaim -n -i -d "REDUXIO TCAS"  
Success, reboot required.
```



A reboot is required after successful completion.

## Load Balancing Policy

The recommended load balancing policy for the Reduxio HX Series system is Round-Robin with Subset. This policy performs round-robin on the Active/Optimized paths. Other paths are attempted on a round-robin basis if all Active/Optimized paths fail.

To set the default load balancing policy to Round-Robin with Subset launch PowerShell and type `Set-MSDSMGlobalDefaultLoadBalancePolicy -Policy RR` at the PowerShell prompt.

```
PS C:\> Set-MSDSMGlobalDefaultLoadBalancePolicy -Policy RR
```

## Required Registry Settings

The table below describes the Windows Registry settings required for proper high-availability operations with Reduxio HX Series systems. These settings are located in the following registry key:

`HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\000x\Parameters` where x is a single digit.

## Required Windows Registry Settings

<b>Key</b>	<b>Value</b>	<b>Explanation</b>
NewPDORemovePeriod	360	Specifies a physical device object (PDO) removal period, in seconds. This period is the length of time the server waits after all paths to a PDO have failed before it removes the PDO.
NewPathVerificationPeriod	30	Specifies a path verification period, in seconds. This is the length of time for the server to verify every path. This parameter is not relevant unless the path verification state has a value of Enabled.
NewRetryCount	120	Specifies the number of times to retry an I/O request.
NewRetryInterval	3	Specifies a retry interval, in seconds. This is the length of time after which the server retries a failed I/O request.
NewDiskTimeout	60	Specifies the disk timeout value, in seconds. This value is the length of time the server waits before it marks the I/O request as timed out.
LinkDownTime	90	This value determines how long requests will be held in the device queue and retried if the connection to the target is lost.
MaxRequestHoldTime	90	Maximum time (in seconds) for which requests will be queued if connection to the target is lost and the connection is being retried. After this hold period, requests are failed with error no device and the device (disk) is removed from the system.

Type the following commands in PowerShell to manually set the required parameters:

```
Set-MPIOSetting -NewPDORemovePeriod 360 -PathVerificationState Enabled \
-NewPathVerificationPeriod 30 -NewRetryCount 120 -NewRetryInterval 3 -NewDiskTimeout
60

$RegistryIscsiParametersPath0 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\
{4d36e97b-e325-11ce-bfc1-08002be10318}\0000\Parameters"
$RegistryIscsiParametersPath1 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\
{4d36e97b-e325-11ce-bfc1-08002be10318}\0001\Parameters"
$RegistryIscsiParametersPath2 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\
{4d36e97b-e325-11ce-bfc1-08002be10318}\0002\Parameters"
$RegistryIscsiParametersPath3 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\
{4d36e97b-e325-11ce-bfc1-08002be10318}\0003\Parameters"
$RegistryIscsiParametersPath4 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\
{4d36e97b-e325-11ce-bfc1-08002be10318}\0004\Parameters"

if ( Test-Path $RegistryIscsiParametersPath0 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath0"
}
if ( Test-Path $RegistryIscsiParametersPath1 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath1"
}
if ( Test-Path $RegistryIscsiParametersPath2 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath2"
}

if ( Test-Path $RegistryIscsiParametersPath3 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath3"
}

if ( Test-Path $RegistryIscsiParametersPath4 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath4"
}

Set-ItemProperty -Path $RegistryIscsiParametersPath -Name LinkDownTime -Value
$NewLinkDownTime
Set-ItemProperty -Path $RegistryIscsiParametersPath -Name MaxRequestHoldTime -Value
$NewMaxRequestHoldTime
```

# Validation

---

This section includes the following topics:

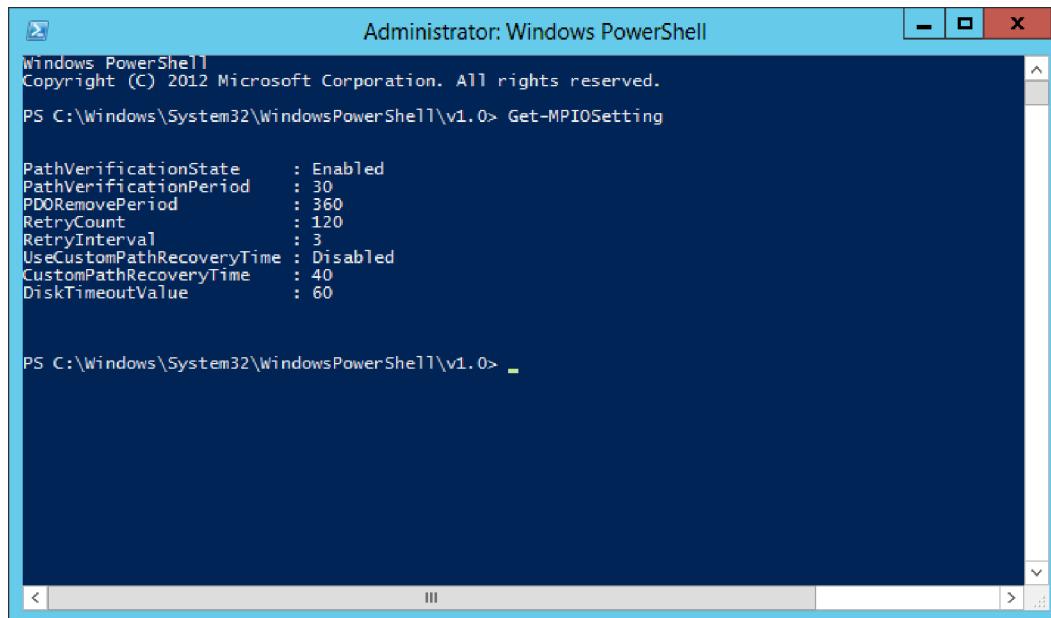
---

<b>Validating Windows Registry Settings .....</b>	<b>10</b>
<b>Validating Microsoft Multipath I/O .....</b>	<b>11</b>

## Validating Windows Registry Settings

To validate the Windows Registry settings configured using the `Set-MPIOSetting` command, launch PowerShell and type `Get-MPIOSetting` at the prompt. Validate that the settings match the ones documented in [Required Registry Settings](#) on page 7.

An example output with the correct settings is shown below:



The screenshot shows a Windows PowerShell window titled "Administrator: Windows PowerShell". The window is running on Windows Server 2012 R2. The command `Get-MPIOSetting` was run, and the output shows the following registry settings:

```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

PS C:\Windows\System32\WindowsPowerShell\v1.0> Get-MPIOSetting

PathVerificationState      : Enabled
PathVerificationPeriod    : 30
PDRemovePeriod           : 360
RetryCount                : 120
RetryInterval             : 3
UseCustomPathRecoveryTime : Disabled
CustomPathRecoveryTime    : 40
DiskTimeoutValue          : 60

PS C:\Windows\System32\WindowsPowerShell\v1.0> -
```

The LinkDownTime and MaxRequestHoldTime parameters cannot be updated using the Set-MPIOSetting command. To validate these settings, launch PowerShell and type the following commands at the prompt:

```
$R = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}"

$RegistryIscsiParametersPath0 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\000\Parameters"
$RegistryIscsiParametersPath1 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\0001\Parameters"
$RegistryIscsiParametersPath2 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\0002\Parameters"
$RegistryIscsiParametersPath3 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\0003\Parameters"
$RegistryIscsiParametersPath4 = "HKLM:\SYSTEM\CurrentControlSet\Control\Class\{4d36e97b-e325-11ce-bfc1-08002be10318}\0004\Parameters"

if ( Test-Path $RegistryIscsiParametersPath0 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath0"
}
if ( Test-Path $RegistryIscsiParametersPath1 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath1"
}
if ( Test-Path $RegistryIscsiParametersPath2 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath2"
}

if ( Test-Path $RegistryIscsiParametersPath3 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath3"
}
if ( Test-Path $RegistryIscsiParametersPath4 ) {
    $RegistryIscsiParametersPath = "$RegistryIscsiParametersPath4"
}

$LinkDownTime = (Get-ItemProperty -Path $RegistryIscsiParametersPath -Name LinkDownTime -ErrorAction SilentlyContinue).LinkDownTime
$MaxRequestHoldTime = (Get-ItemProperty -Path $RegistryIscsiParametersPath -Name MaxRequestHoldTime -ErrorAction SilentlyContinue).MaxRequestHoldTime

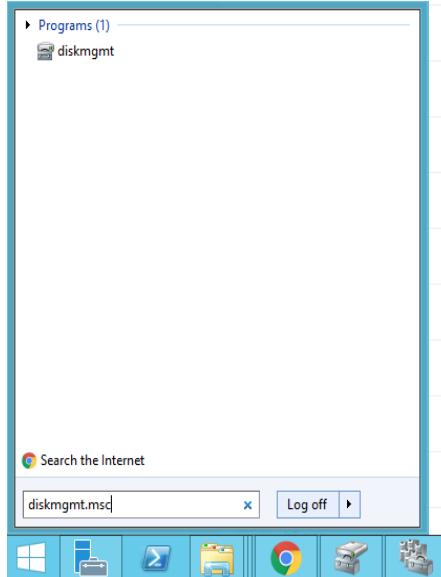
Write-Host "LinkDownTime=$LinkDownTime"
Write-Host "MaxRequestHoldTime=$MaxRequestHoldTime"
```

## Validating Microsoft Multipath I/O

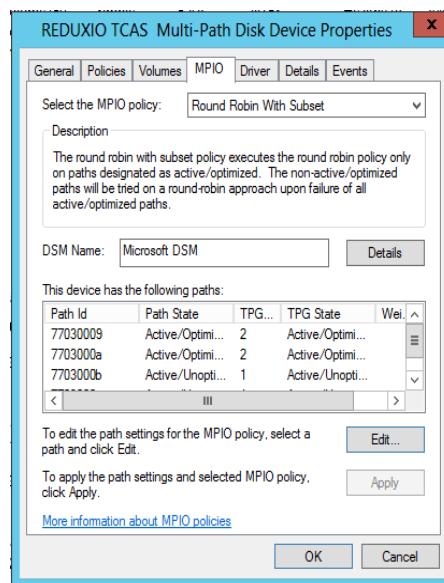
To verify that multipathing is working as expected, identify the attached Reduxio disk and review its properties, as follows:

1. Right-click **Start** and then click **Search**.

2. Type `diskmgmt.msc` in the search field. Click `diskmgmt.msc` in the results pane to open **Disk Management**.



3. Right-click **Reduxio disk** to view its properties.
4. Click the **MPIO** tab.



- 
- 5.** Verify that there are at least two Active/Optimized paths and two Active/Unoptimized paths.
-

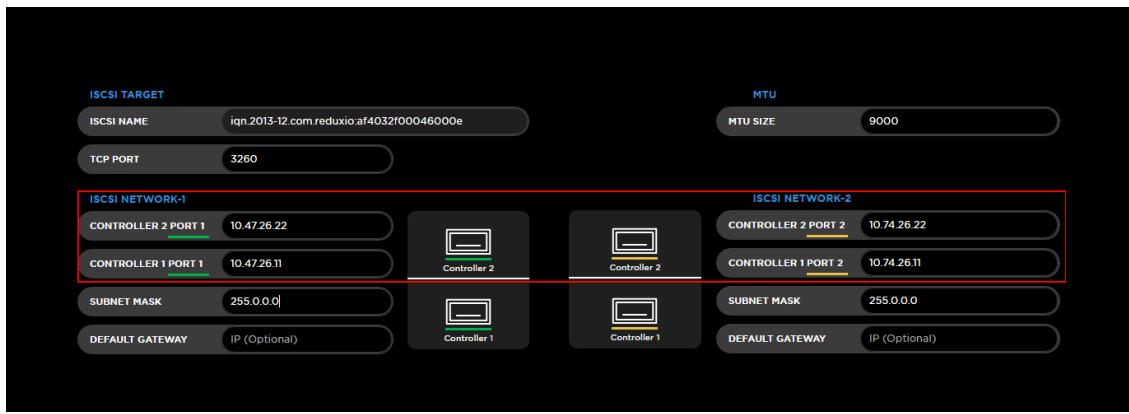
# Manual Multipath Configuration

---

Multipath configurations are applied through the Reduxio StorKit for Microsoft Windows Server script. Follow this procedure to manually configure multipathing.

## Identify Reduxio iSCSI IP Addresses

1. In Reduxio Storage Manager click **Settings** from the icon bar.
2. Click **NETWORK CONFIGURATION** to view data interface IP addresses.
3. Write down the four iSCSI IP addresses listed in the **CONTROLLER 1 & 2** and the **PORT 1 & 2** fields for further use.

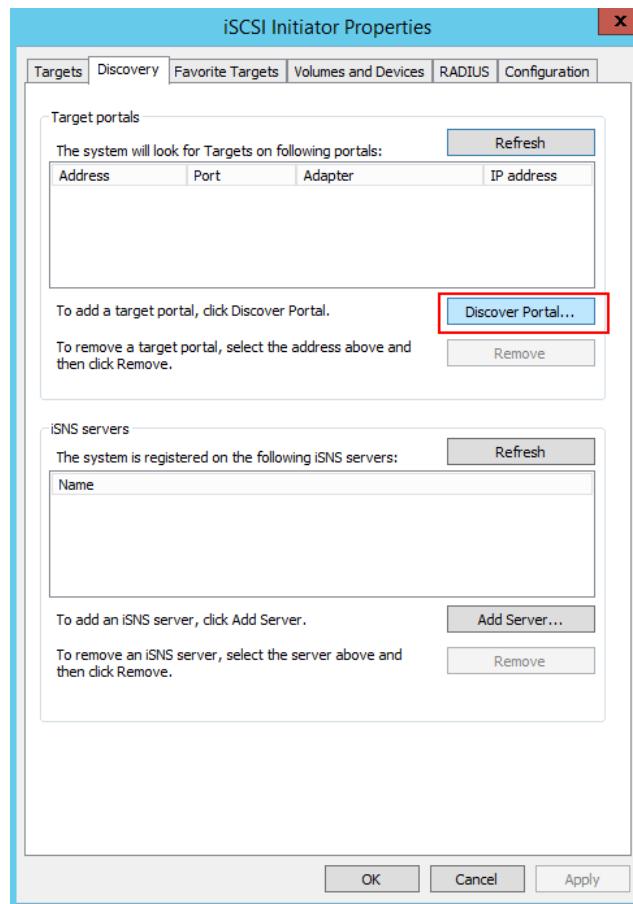


## Configure iSCSI Initiator

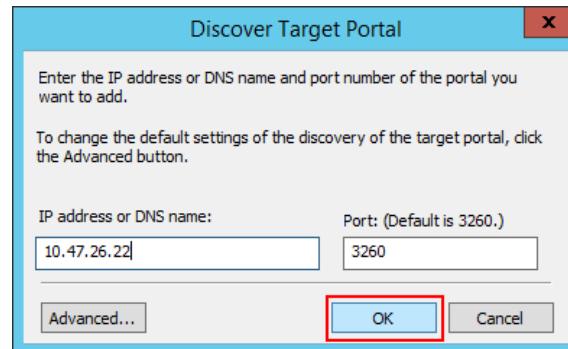
Use RDP to initially configure the iSCSI initiator on a windows 2012 R2 server (or higher) to connect to Reduxio storage, as shown in the following procedure:

1. Click **Control Panel > iSCSI Initiator**.

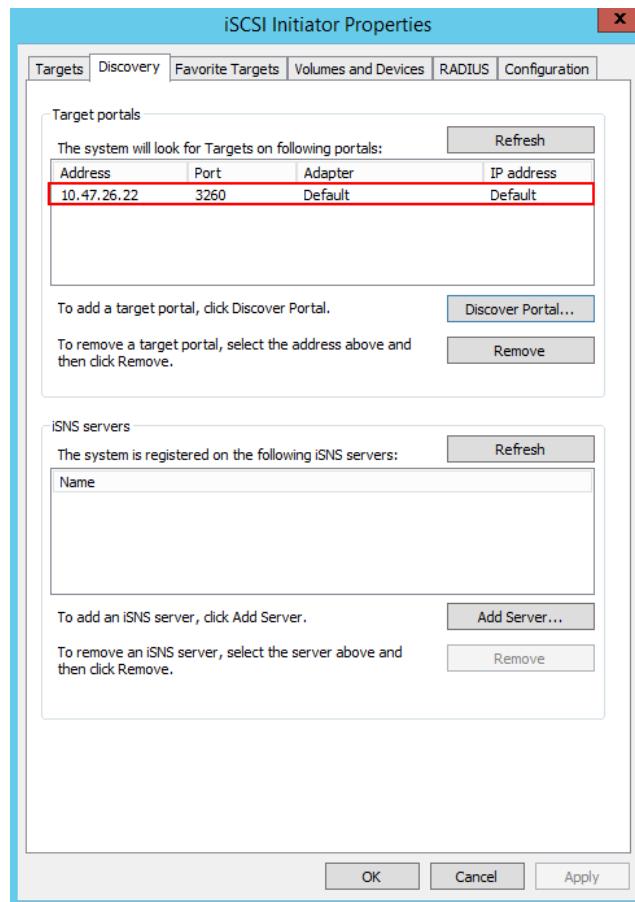
2. In the **Discovery** tab click **Discover Portal**.



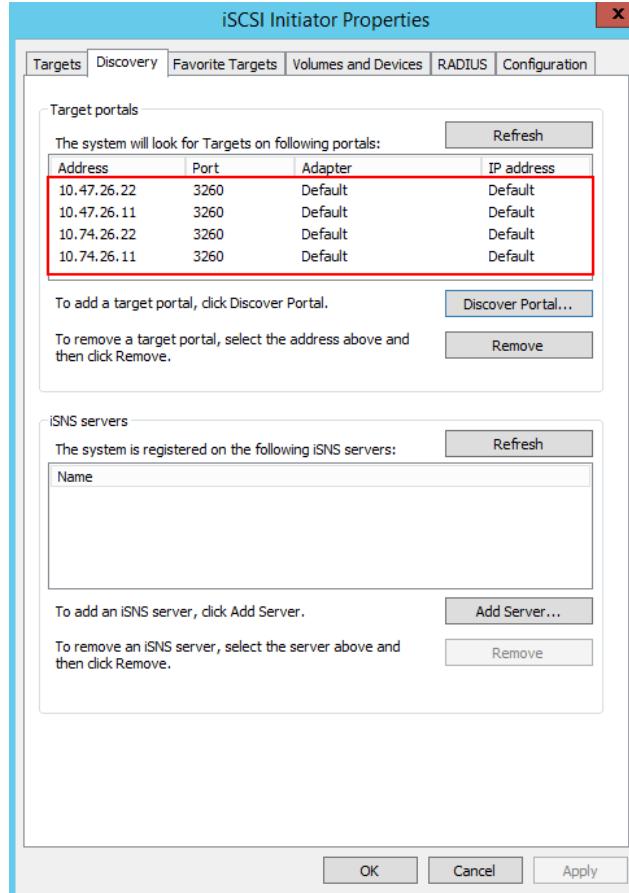
3. Type the data IP of the first target identified in [Identify Reduxio iSCSI IP Addresses](#) (page 14) and click **OK**.



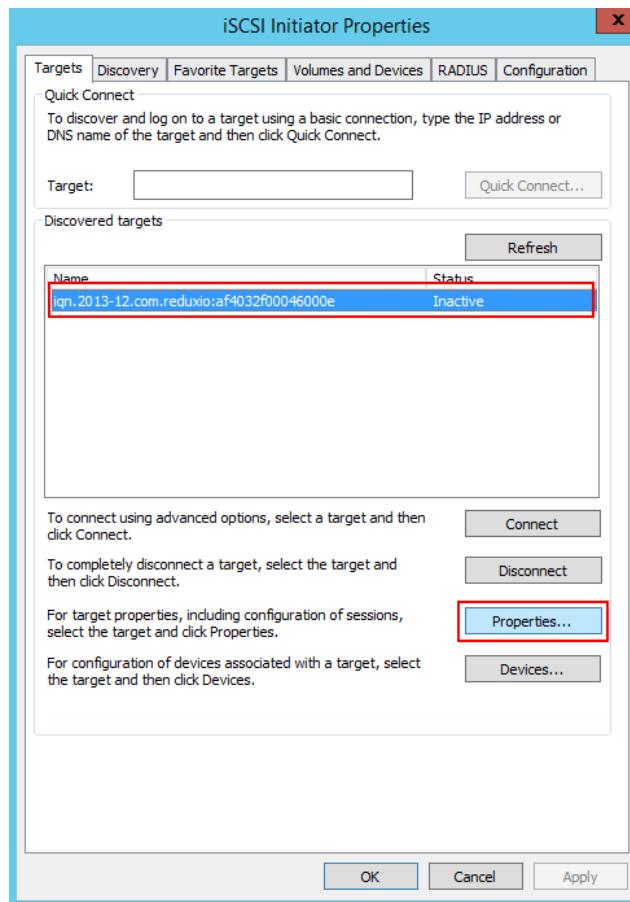
4. Click the **Discovery** tab and verify that this new target appears in the Target portals.



5. Repeat steps 2-4 for the data IPs of the other three targets identified in [Identify Reduxio iSCSI IP Addresses](#) (page 14). When you are done verify that all four targets appear in the Target portals in the **Discovery** tab.

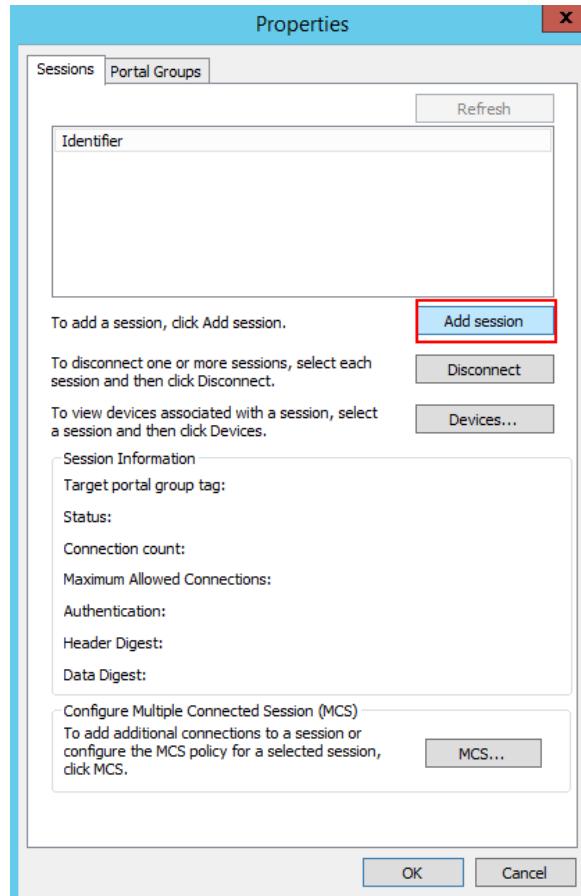


- 
6. In the **Targets** tab click the discovered target and then click **Properties**.

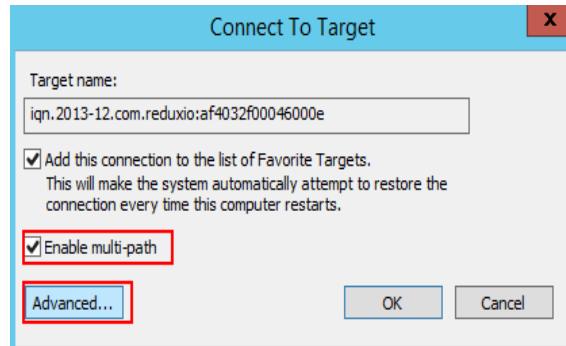


## Configure Multipathing

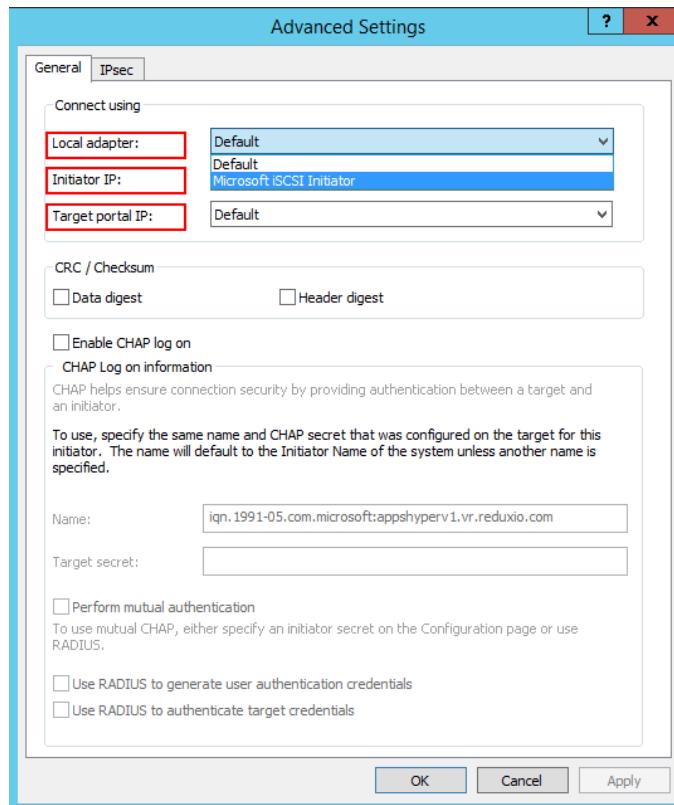
1. In the **Sessions** tab of the **Properties window** click **Add Session**.



- 
2. In the **Connect to Target** window select **Enable multi-path** and click **Advanced**.



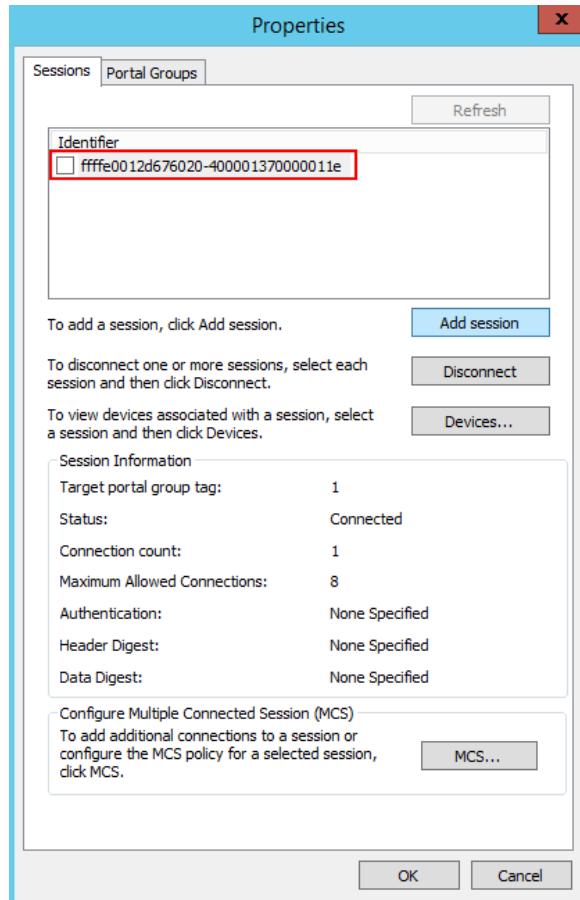
3. In the **Advanced Settings** window click the **General** tab and make the following selections from the three drop-down lists in the **Connect Using** pane:
- **Local Adapter: Microsoft iSCSI Initiator**
  - **Initiator IP:** your first iSCSI NIC
  - **Target portal IP** your first IP



This step is done three additional times for your first iSCSI NIC. During each subsequent step select the next IP in the **Target portal IP** drop-down list until all four have been selected.

4. Click **OK** to close the **Advanced Settings** window.

5. In the **Connect to Target** window click **OK**. Verify that the new target name appears in the **Sessions** tab of the **Properties** window.

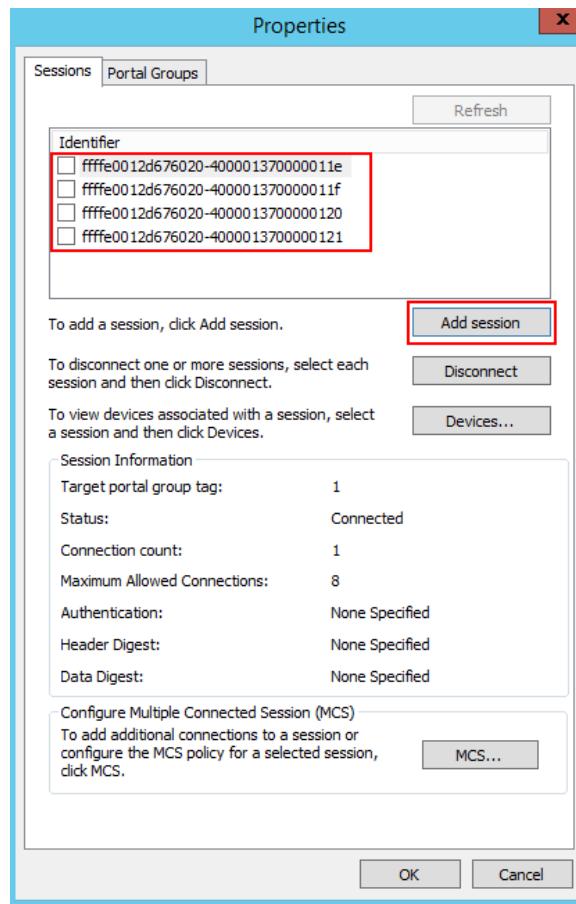


6. Repeat steps 2-5 for the data IP of the other three targets for the first iSCSI NIC.

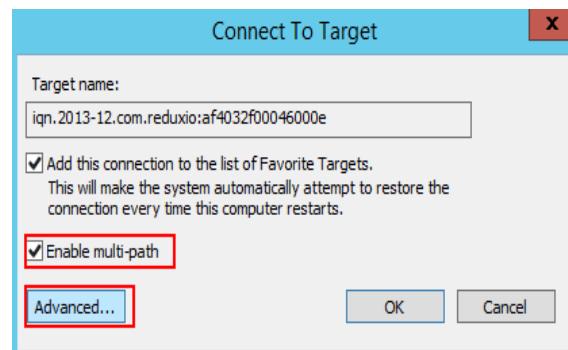


In each iteration, keep the the Initiator IP the same but select a different Reduxio data network IP as the Target Portal IP. In configurations with multiple host NIC ports repeat these steps again with the additional Initiator IP.

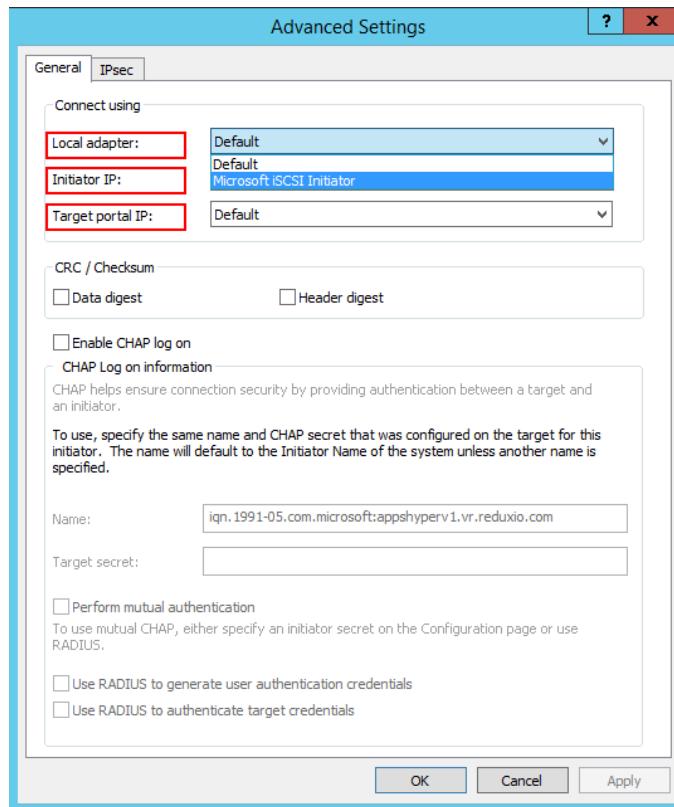
7. When you see four sessions in the **Properties** screen click **Add Session**.



8. In the **Connect to Target** window select **Enable multi-path** and click **Advanced**.



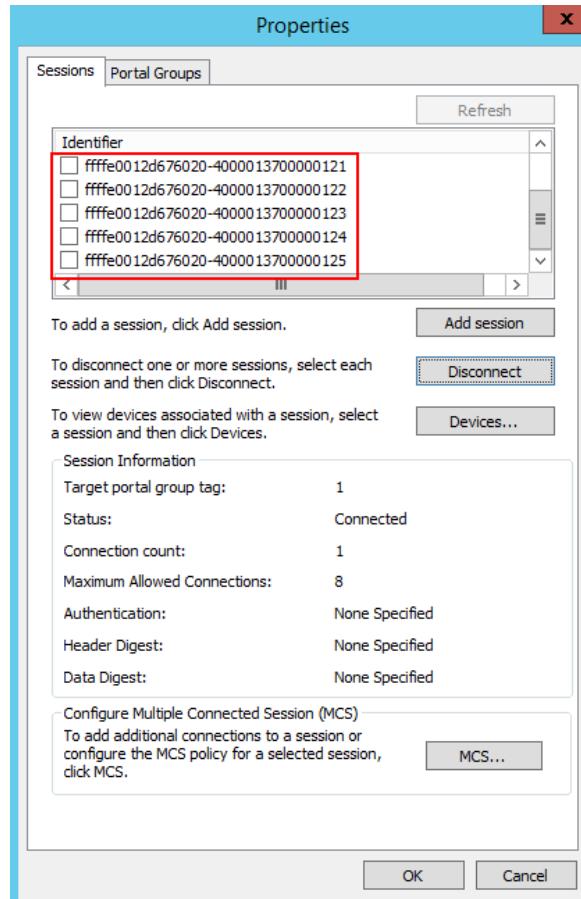
9. In the **Advanced Settings** window click the **General** tab and make the following selections from the three drop-down lists in the **Connect Using** pane:
- **Local Adapter: Microsoft iSCSI Initiator**
  - **Initiator IP:** your second iSCSI NIC
  - **Target portal IP** your first IP



This step is done three additional times for your second iSCSI NIC. During each subsequent step select the next IP in the **Target portal IP** drop-down list until all four have been selected.

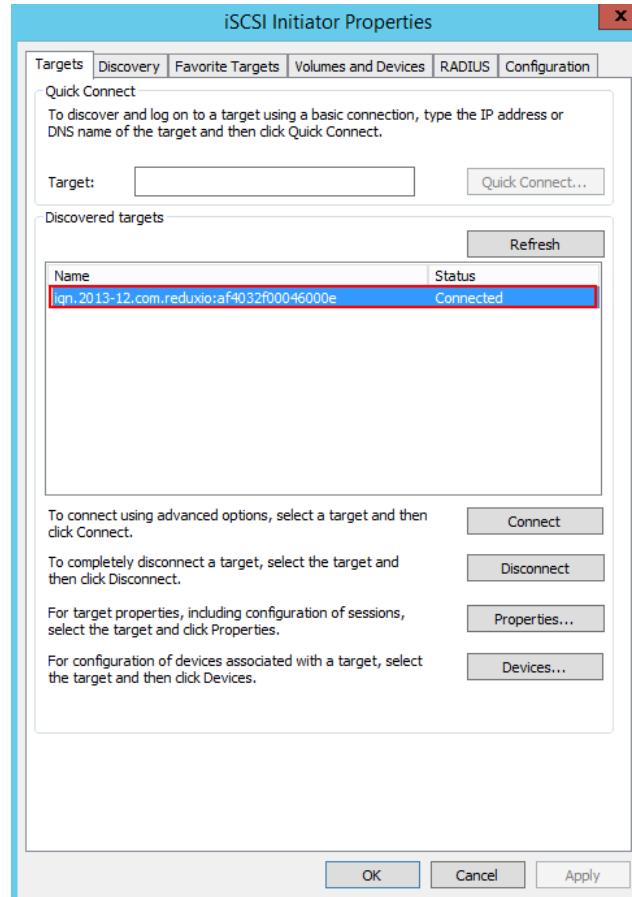
10. Click **OK** to close the **Advanced Settings** window.
11. In the **Connect to Target** window click **OK**. Verify that the new target name appears in the **Sessions** tab of the **Properties** window.
12. Repeat steps 8-11 for the data IP of the other three targets for the second iSCSI NIC.

13. When you see eight sessions in the **Properties** screen click **OK**.



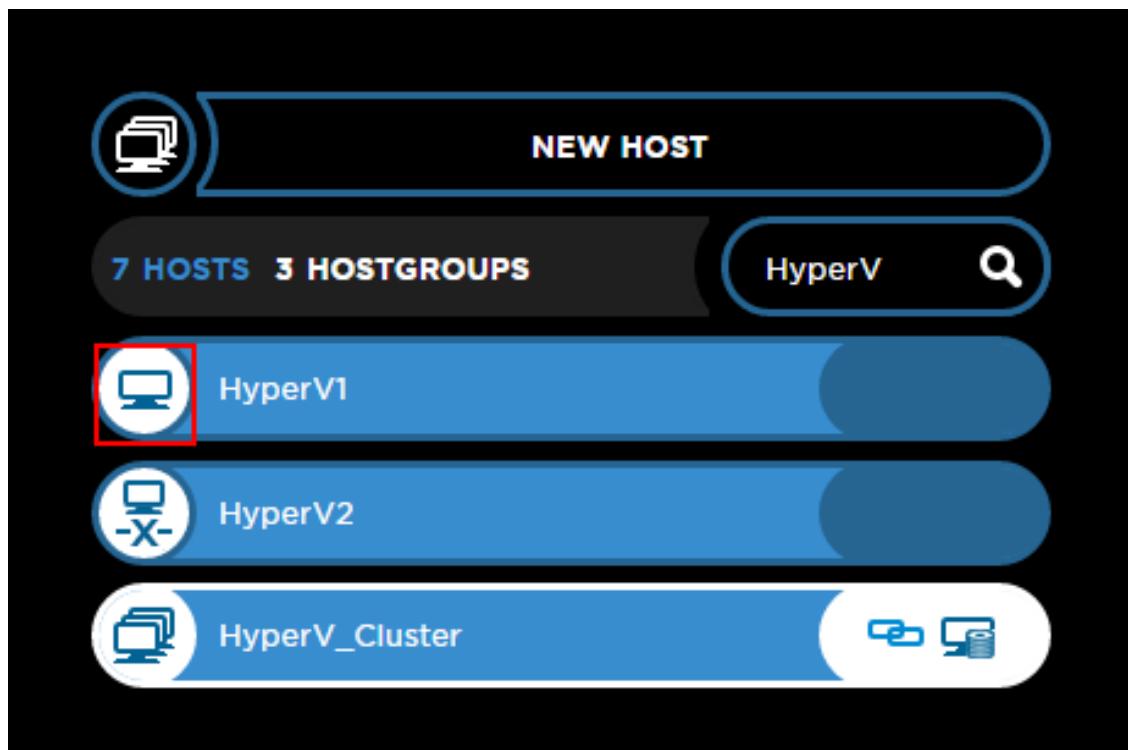
## Validating Multipathing Connection

1. Click the **Targets** tab of the **iSCSI Initiator Properties** window and verify that the status of the discovered target is **Connected**.

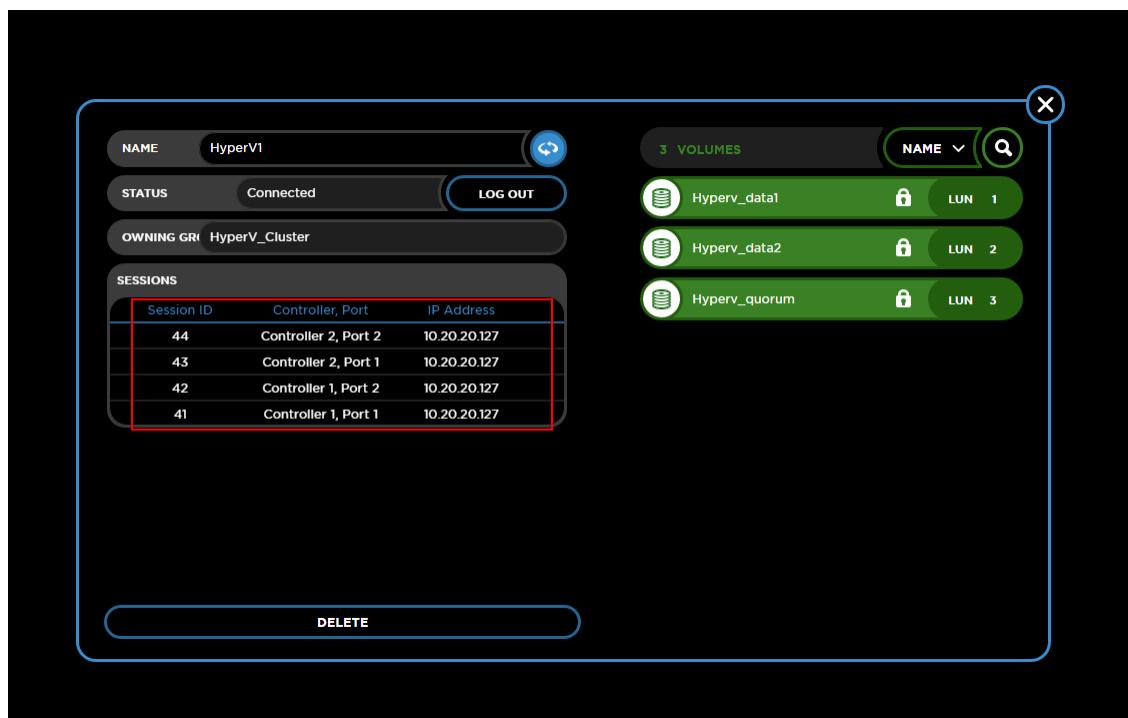


2. In the Reduxio Storage Manager click the **HOSTS & VOLUMES** icon in the icon bar.
3. Type the properties of the server defined with the multipathing configuration.

4. Verify that the multipathing server is connected.



5. Click **SESSIONS** and verify that all eight sessions are available.



# Conclusions

---

The Reduxio StorKit for Microsoft Windows Server provides a PowerShell script to configure best practice settings required for proper high availability in Windows Server environments. It is important to perform these changes in hosts connected to Reduxio HX Series systems.

## Reduxio Documentation

- [Reduxio Support Portal](#): *Reduxio TimeOS Administration Guide*

## Microsoft Documentation

- [TechNet](#): Microsoft Multipath I/O Step-by-Step Guide
- [MSDN Blog](#): Updated Guidance on Microsoft MPIO Settings
- [TechNet](#): Set-MPIOSetting