



WHITE PAPER

GUIDE TO ISCSI PERFORMANCE TESTING ON THE CISCO MDS 9000 FAMILY

INTRODUCTION

This document provides an easy-to-follow guide for building a Small Computer System Interface over IP (iSCSI) implementation using the Cisco® MDS 9000 Series of multiprotocol switches. Specifically, this document will focus on a quick, simple setup for configuring a Cisco MDS 9000 Series IP services module for the purposes of conducting iSCSI proof-of-concept and validation tests. Below are the four steps needed to accomplish this:

Note: Command-line interface configurations will be shown in Appendix B.

- Cisco MDS 9000 Series baseline configuration
 - Enable Gigabit Ethernet interface
 - Set iSCSI authentication to **None**
 - Configure Gigabit Ethernet interface for iSCSI
 - Set iSCSI mode to **Store and Forward**
- Configuring access for iSCSI clients on Cisco MDS 9000 Series
 - Create iSCSI initiator by IP address
 - Enable **Dynamic Import Target FC**
 - Zone Fibre Channel targets and iSCSI initiator
- Installing and configuring iSCSI client (Microsoft driver)
 - Log into Cisco MDS 9000 Series Gigabit Ethernet interface
 - Log into available iSCSI targets
- Using the Iometer performance test tool
 - Test parameters affecting performance

Note: It is important that the Ethernet management port on the Cisco MDS 9000 Series switch be configured on a separate VLAN and IP subnet from the Gigabit Ethernet ports on the IP services module. This requirement is mandatory and is required to make sure the management port does not interfere with the IP subnet(s) configured on the IP services module (gratuitous Address Resolution Protocol [ARP] requests).

TOPOLOGY INFORMATION

The following configuration is used as an example configuration and serves as a baseline to gauge performance results. Although it might not be feasible to use exactly the same server configuration, the results presented in this paper can be compared with any results gained from using different server hardware to gauge the validity of the data.

Host Information

IBM x345 PC

- Dual Xeon 2.66-GHz CPU
- 1 GB of memory
- Windows 2000 Advanced Server with Service Pack 4
- Microsoft iSCSI driver Version 1.0.4a
- Dual Intel PRO/1000 MT network card with driver Version 7.3.13 (dated 10/28/2003)
- Test tool: Iometer Version 2003.12.16

IP Switch Information

Cisco Catalyst® 6500 Series Switch

- Software Version 6.3(10)
- 8 optical Gigabit Ethernet interfaces
- 16 copper Gigabit Ethernet interfaces

Cisco MDS 9000 Series Switch Information

Cisco MDS 9509 Multilayer Director Switch

- One 16-port Fibre Channel switching module (DS-X9016)
- One IP services switching module (DS-X9308SMIP)
- Software Version 1.3(4b)

Storage Information

EMC Clariion CX600

- Navisphere revision 6.4.0.5.2, Generation 116
- 4 GB of cache per controller
- Eight RAID5 LUNs—5 GB each

CISCO CATALYST ETHERNET SWITCH CONFIGURATION

The ports that need to be configured are the Cisco Catalyst switch Ethernet ports that the iSCSI host and Gigabit Ethernet port on the Cisco MDS 9000 Series IP services module are connected to. On the Cisco Catalyst Ethernet switch, a few basic configurations need to be conducted on these ports:

- Cisco EtherChannel® capability and VLAN trunking should be disabled; this is a normal configuration for “host” ports and reduced port bring-up times as a result of reduced protocol negotiation
- PortFast should be enabled; this is an Ethernet feature from Cisco Systems® that reduces the port bring-up time as a result of optimizing 802.1D Spanning Tree timers specific to “host” ports

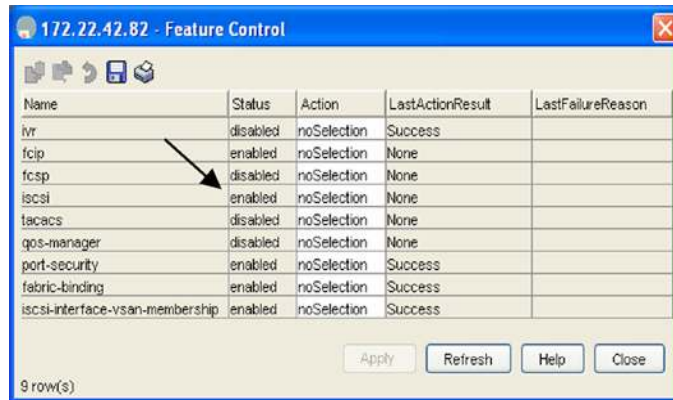
For the purposes of creating the iSCSI test environment, it is recommended to keep the configuration simple and assign an IP subnet that is to be used for both the iSCSI initiator(s) and the Cisco MDS 9000 Series IP services module’s Gigabit Ethernet port. If using multiple Cisco MDS 9000 Series IP services module ports, assign a separate IP subnet for each Gigabit Ethernet port that will also be used by the iSCSI initiators assigned to that port.

CISCO MDS 9000 SERIES BASELINE CONFIGURATION

On the Cisco MDS 9000 Series, some basic configurations are needed for iSCSI.

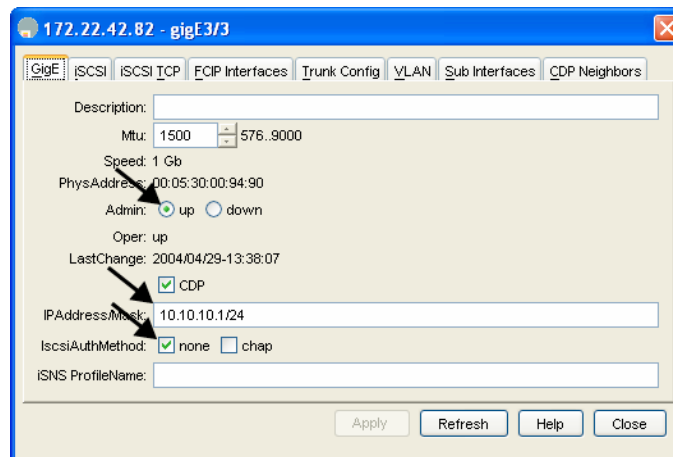
Step 1—Enable the iSCSI feature (global command) on the Cisco MDS 9000 Series switch.

- 1) From the Device Manager, click the toolbar **Admin → Feature Control**.
- 2) Verify that iSCSI is enabled. If it is not, click the drop-down box under Action and select **enabled**.



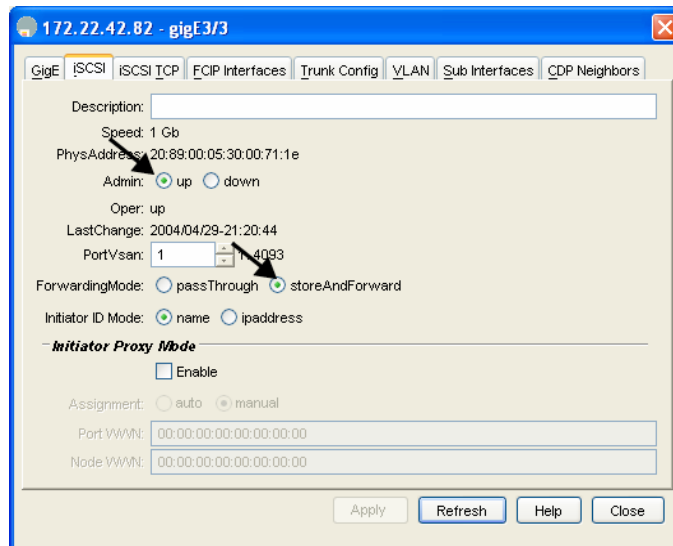
Step 2—Enable and configure the IP services module Gigabit Ethernet port.

- 1) From the Device Manager, right-click the Gigabit Ethernet interface and select **Configure**.
- 2) Enable the interface and assign an IP address and subnet mask.
- 3) Set the iSCSI authentication mode to **none**.



Step 3—Enable and configure iSCSI on the IP services module Gigabit Ethernet port

- 1) From the same configuration window, click the **iSCSI** tab, which is the second tab from the left.
- 2) Enable the corresponding iSCSI logical interface.
- 3) Set the iSCSI Forwarding Mode to **Store And Forward**.

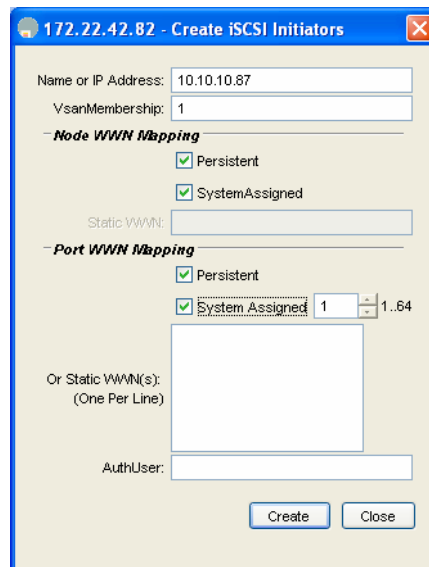


CONFIGURING ACCESS FOR ISCSI CLIENT ON THE CISCO MDS 9000 SERIES

Step 1—Creating the iSCSI initiator

- 1) From the Device Manager, click the toolbar **IP → iSCSI**.
- 2) On the Initiator tab, click **Create**.
- 3) In the **Name or IP Address** window, type in the IP address of the iSCSI host that will be accessing the Cisco MDS Gigabit Ethernet interface; note that if you have two network interface cards (NICs) in the host, please enter the correct NIC IP address that will be accessing the Cisco MDS Gigabit Ethernet for iSCSI.
- 4) In the **VsanMembership** window, enter the VSAN number where the storage is located; this is where the iSCSI initiator will be accessing.
- 5) In **Node WWN Mapping** and **Port WWN Mapping**, select both **Persistent** and **System Assigned** check boxes.
- 6) In **Port WWN Mapping** beside the System Assigned check box, leave the entry as 1.
- 7) Click **Create** and then click **Close**.

A screen shot of this procedure follows:

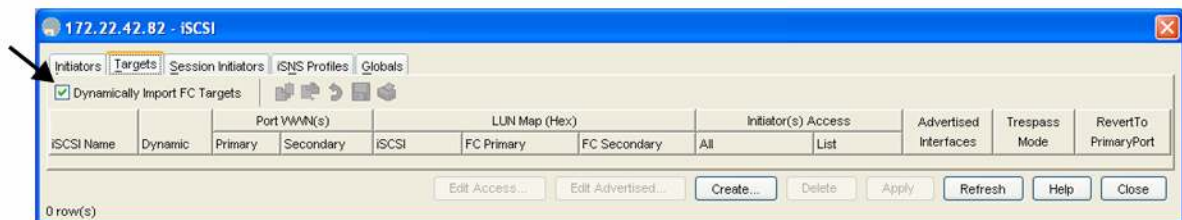


Step 2—Enabling the **Dynamic Import FC Targets**

This feature allows you to auto-import Fibre Channel targets as iSCSI targets

- 1) From the Device Manager, click the toolbar **IP → iSCSI**. A window will appear; click the second tab, called *Targets*.
- 2) In the check box called **Dynamically Import FC Targets**, add a check.

A screen shot follows:



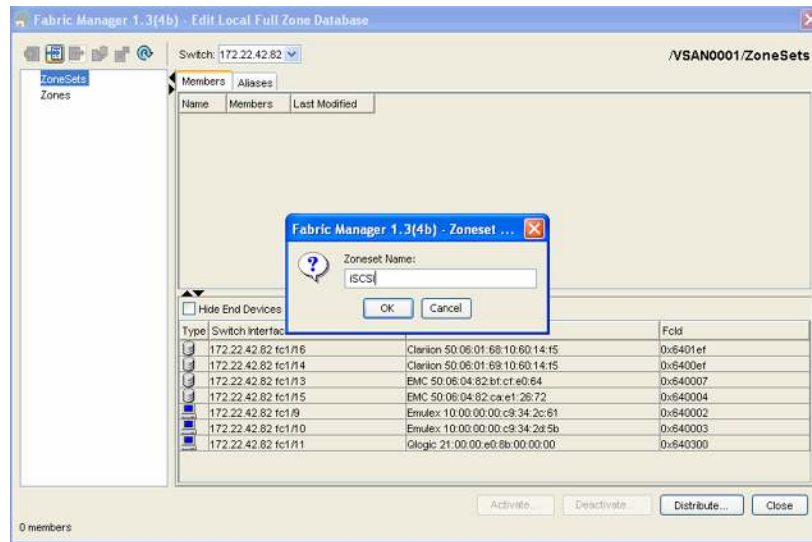
Like any other normal Fibre Channel host, an iSCSI client will need to be granted permission to communicate with the storage device. This permission is granted through zoning. Below are steps to add storage and an iSCSI client into a zone.

Step 3—Zoning Fibre Channel targets and iSCSI initiator

From the Fabric Manager, select your iSCSI VSAN, and from the toolbar, click **Zone → Edit Local Full Zone Database**.

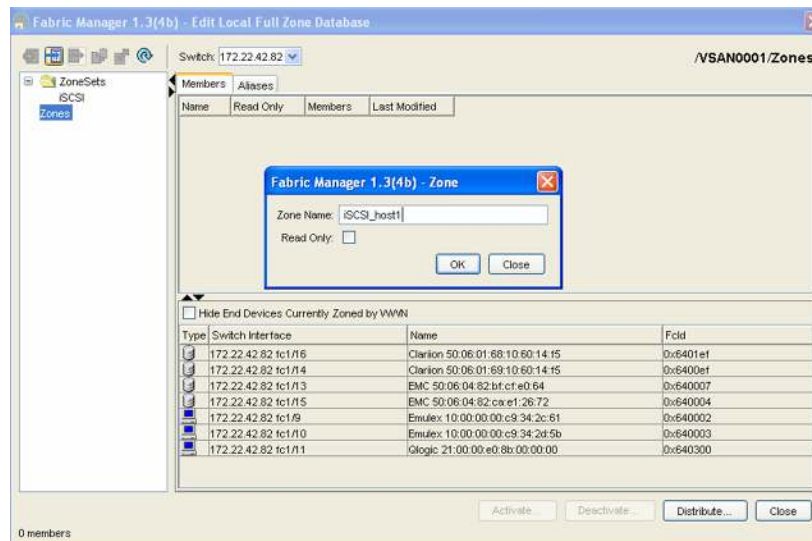
If you do not have a ZoneSet already, create one.

- 1) To create a new ZoneSet, highlight ZoneSets on the left-hand panel and click the right blue arrow.
- 2) Type in a name for your ZoneSet, then click **OK**; default is ZoneSet1. In the example below, the ZoneSet name is **iSCSI**.



When ZoneSet creation is completed, select **Zones** on the left-hand panel and click the right blue arrow.

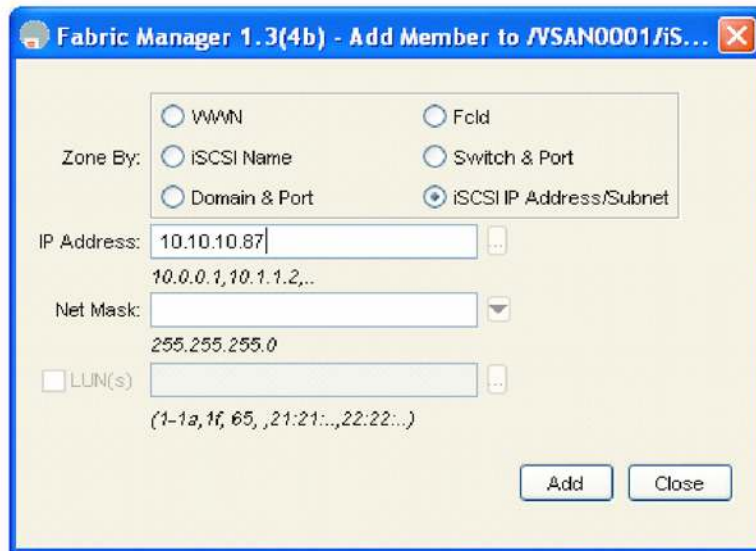
- 1) A window will appear, and you will need to enter the name of the zone; once completed, click **OK**; the default name is Zone1.
- 2) In the example below, the Zone Name is iSCSI_host1.



To add storage members to the new zone that was created, select the new zone name on the left-hand panel.

- 1) Drag and drop your Fibre Channel storage into the zone.
- 2) Once you have completed adding all the storage ports into the zone, select the zone and click the blue right-arrow.
- 3) Select the radio button **iSCSI IP Address/Subnet**.
- 4) In the IP Address window, enter the IP address of the iSCSI initiator and click **Add** and then click **Close**.

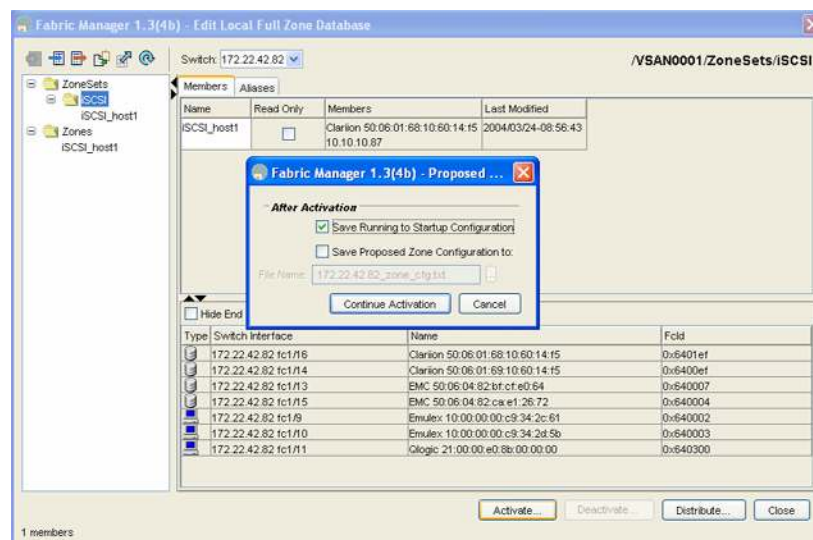
A screen shot of this task follows:



You will now need to add the zone to the ZoneSet and then activate the ZoneSet, following these steps:

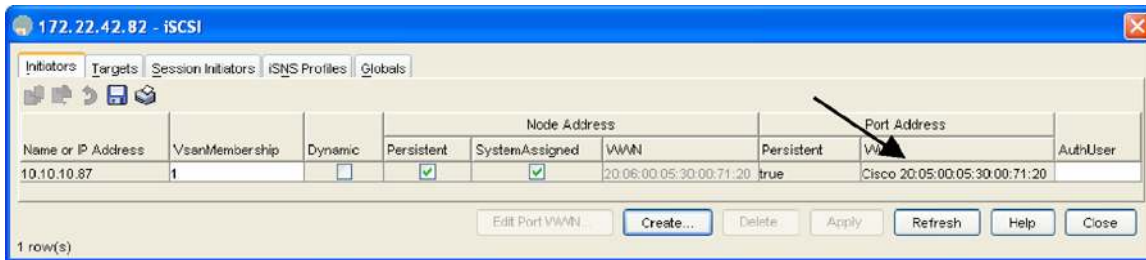
- 1) After you have completed creating the zone, drag and drop this new zone into your ZoneSet.
- 2) Highlight the ZoneSet (iSCSI) and click **Activate**.
- 3) You will be prompted to save the configuration; click **Continue Activation**.

A screen shot of this task follows:



Because the storage array used in this example is an EMC Clariion array with embedded LUN security, the port world wide name (pWWN) of the iSCSI client is needed in order to grant access to the iSCSI client to the Clariion storage (LUNs). To view the pWWN of the iSCSI initiator, go to the Device Manager **under IP->iSCSI**, and view the **Initiators** tab as shown below. This pWWN needs to be entered into the

Clariion's **Connectivity Status** window and will need to be registered and configured to see LUNs assign to Storage Groups. For more detail configuration, please contact your EMC engineer for assistance.



Note: This step of configuring LUN security is required only when using a storage array with LUN security enabled. If using just a bunch of disks (JBOD) or other disk array without LUN security, you may skip this step.

INSTALLING MICROSOFT'S iSCSI DRIVER

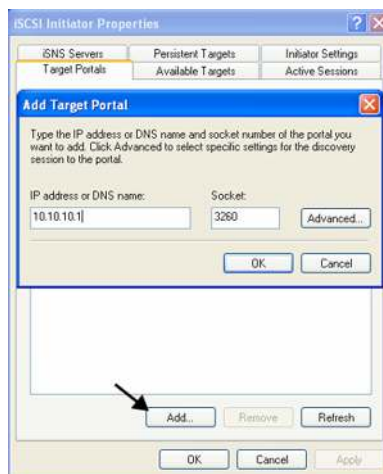
The Ethernet NIC on the iSCSI initiator server must be connected to the Cisco Catalyst Ethernet switch and must be assigned an IP address within the designated IP subnet. As a test, use the Windows IP Ping CLI utility to verify that the Cisco MDS 9000 Series IP storage port can be pinged from the server before proceeding.

The next step is to download the latest version of Microsoft's iSCSI driver at Microsoft's iSCSI page:

<http://www.microsoft.com/windowsserversystem/storage/technologies/iscsi/default.mspx>.

Once you have installed the Microsoft iSCSI driver and previously verified IP connectivity to the Cisco MDS 9000 Series IP services Gigabit Ethernet port, establishing an iSCSI session to the Cisco MDS 9000 Series IP services module is the next step.

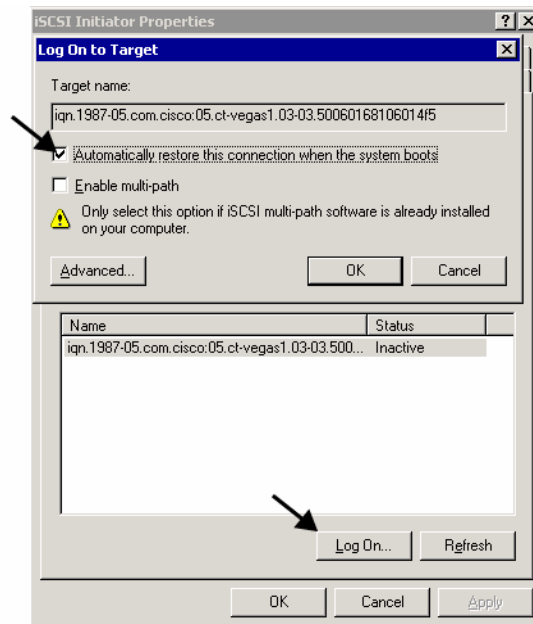
The Microsoft iSCSI GUI will have an icon on your desktop once it is installed. To run the iSCSI configuration, double-click this icon. Once in, click the **Target Portals** tab in the iSCSI Initiator Properties dialog window and then click **Add**. Enter the IP address of the Cisco MDS 9000 Series IP services module Gigabit Ethernet port. Leave the Socket number as the default, 3260.



Upon clicking **OK**, an iSCSI session will be established. Once connection to the Target Portal is established, this iSCSI initiator will have an entry created in the Fibre Channel Name Server (fcns) database. The Cisco MDS 9000 Series will assign a node world wide number (nWWN) and pWWN with an associating FCID to this iSCSI client. This forms the iSCSI-to-Fibre Channel address binding.

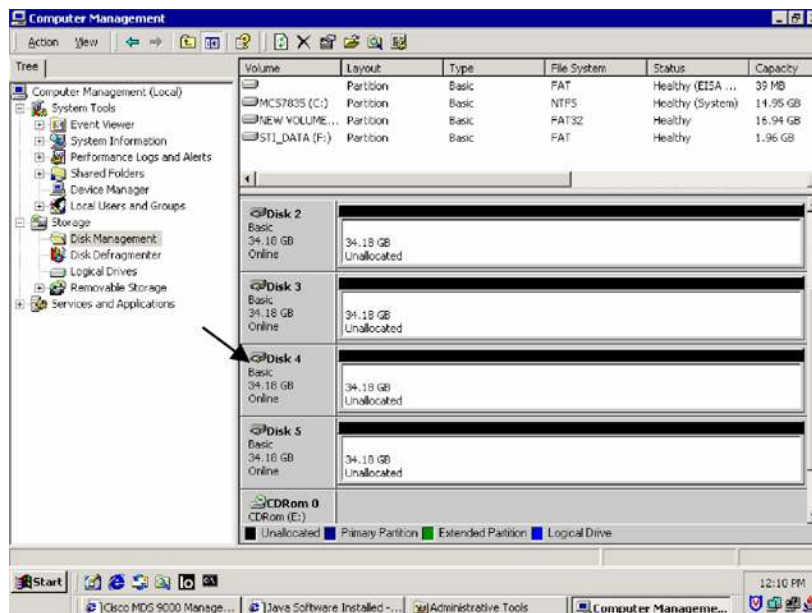
COMPLETING ISCSI CLIENT ACCESS TO STORAGE

On the iSCSI client, open the Microsoft iSCSI driver configuration panel. Under the Available Targets tab, click **Refresh**. The iSCSI target that is associated to the accessible Fibre Channel target device will appear. Once the iSCSI target appears, highlight the entry and click **Log On....** Check the box **Automatically restore this connection when the system boots**, then click **OK**. A screen shot of this action follows:



At this point, the iSCSI client will have access to the assigned Fibre Channel disk. In some cases, you will need to go to **Disk Management** under the Microsoft Window's Computer Management window to initiate a rescan of the disk devices for Windows to recognize and assign a device ID to the newly discovered disk.

You will need to format these new iSCSI disks. Please remember that the Microsoft iSCSI driver does not support Dynamic disks. So when creating a new volume from the iSCSI disk, make sure that the disk(s) are Basic disks. A screen shot of what the disk(s) should look like follows. If they are Dynamic, you can reset them to Basic by right-clicking the disk and selecting **Revert back to Basic**.



USING THE IOMETER PERFORMANCE TEST TOOL

Iometer is a test tool that is used to simulate application input/output (I/O) for the purposes of testing storage and storage networking devices. Using Iometer, one can build I/O profiles that simulate the I/O workload that an application or group of applications might perform. Iometer offers many options, but only a few basic configuration options are needed for the purposes of testing iSCSI.

The two basic configuration tasks are to configure the application, or worker, simulation and to configure the I/O profile or I/O workload to be simulated.

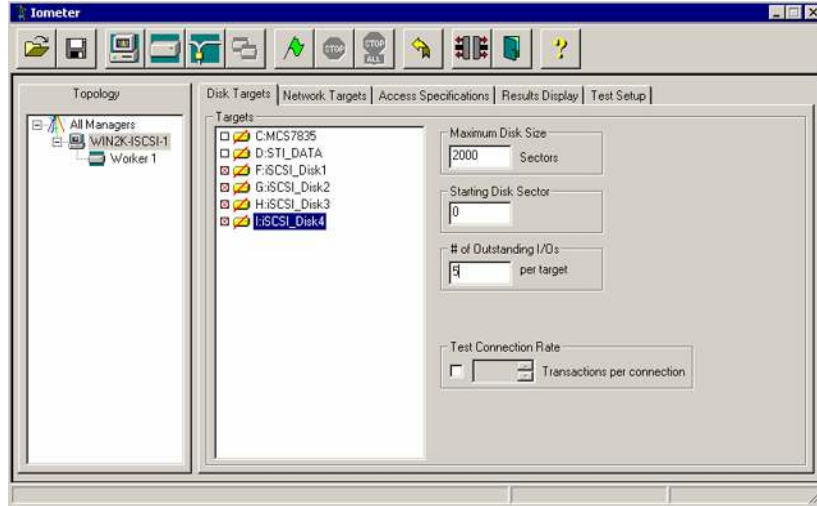
To download the latest Iometer test tool for Windows, go to <http://www.iometer.org/doc/downloads.html>.

Below are some snapshots of the configuration used for the tests results performed for this guide. The parameters for the tests used were:

- 1 worker (one I/O generation agent)
- 2000 sectors = 1 MB of data written to disk
- 5 outstanding I/O (default)

This configuration sets the worker simulation parameters as shown in the dialog below. The number of outstanding I/O requests will affect performance. If you only have one outstanding I/O, it means that the next I/O sent will have to wait until the first I/O is completed. When testing for performance, the storage array will determine how many outstanding I/O requests will give the best performance. Typically this number ranges from 5 to 10, according to these test results.

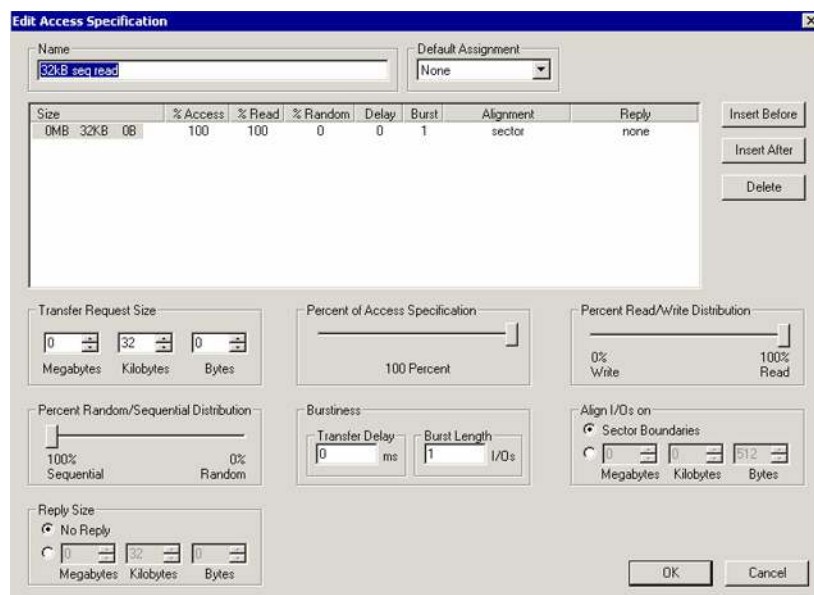
When using the Iometer to test for performance, the more physical disks that the I/O hits, the better the performance. So in case of storage arrays that have RAID capability, the host might see only one LUN but might be actually hitting 10 different physical striped disks as an example. In either case, when running the Iometer test, the more LUNs (drives) hit, the better the performance. Below is a screen shot of the Iometer.



The next step is to configure the I/O profile, or the I/O workload that is to be used to simulate an application. Several parameters can be configured, including I/O size, reads vs. writes, I/O randomness, etc.

Note: Because this is a test of iSCSI and not the disk device itself, it is important to configure these parameters to maximize the performance of the disk array. The goal is to see the performance of iSCSI over a variety of I/O operations (reads/writes) with different block sizes, and therefore it is important not to make the disk array itself a bottleneck. This condition would create inaccurate results. However, it is important to use realistic I/O sizes as well that are representative of the simulated applications. This is discussed later in the document.

The dialog below shows the configuration that was used for this test. The I/O profile consisted of 32 KB read operations that were sequential in nature, thereby maximizing the performance of the disk array.



WHAT I/O PROFILE SHOULD I USE TO SIMULATE MY APPLICATION?

I/O testing tools like Iometer are normally used to simulate actual application I/O patterns (e.g. Microsoft SQL, Microsoft Exchange). Some of these I/O characteristics, including I/O randomness, read or write operations, and block sizes, all depend on the specific application requirements. Customers often ask about the right block size for a specific I/O test. To answer the question, it is important to understand real applications and their respective block sizes in a Windows environment. To help scope the I/O profile configuration, the following section offers some background on Windows-based file systems and applications. Remember, the I/O size relates to a block-based operation only. Even when trying to simulate large file transfers, each file transfer is composed of multiple smaller block-based transactions.

First, most applications perform read and write operations to a Windows file system and not directly to a raw disk device. Windows NT File System (NTFS) is the most common file system in the Windows operating system. Therefore it is necessary to understand the logical block size or cluster size (in Microsoft terminology) of NTFS. In general, smaller cluster sizes are more storage efficient, but they also slow down drive I/O performance because of the high numbers of I/O threads. Table 1 shows the default cluster size of NTFS depending on the partition size (sector = 512 B) for optimal storage efficient and I/O performance.

Table 1
Default Cluster Size of NTFS

Partition Size Range (GB)	Default Number of Sectors Per Cluster	Default Cluster Size (KB)
<= 0.5	1	0.5
> 0.5 to 1.0	2	1
> 1.0 to 2.0	4	2
> 2.0	8	4

Therefore a 4 KB block size should normally be used in an Iometer test to simulate the environment of NTFS read/write with mixed file sizes. For some special applications, people configure up to a 64 KB block (cluster) size for high throughput or streaming purposes (discussed below).

Within Windows NT/2000/XP, it is not possible to use buffered I/O, which results in applications performing direct I/O automatically. This feature actually gives much flexibility to applications to use variable block sizes for different purposes. It means you have the luxury of choosing a relatively larger application block size. There are many advantages to using a relatively large block size. For example, in database applications, one of the most significant advantages is the savings in I/O operations for index-based access paths for database applications such as decision support systems (DSS) sequential data access.

Table 2 describes advantages and disadvantages of different block sizes for database applications and also specifies the range of block sizes for this type of application. In summary, it is obvious that 2 K to 32 K is the most-used block size range in database applications.

Table 2
Block Size Advantages and Disadvantages

Block Size	Advantages	Disadvantages
Small (2K–4K)	Reduces block contention Good for small rows or lots of random access	Has relatively large overhead You may end up storing only a small number of rows, depending on the size of the row
Medium (8K)	If rows are of medium size, you can bring a group of rows into the buffer cache with a single I/O	Space in the buffer cache will be wasted if you are doing random access to small rows and have a large block size

Block Size	Advantages	Disadvantages
Large (16K–32K)	There is relatively less overhead, thus more room to store useful data Good for sequential access, DSS or very large rows	Large block size is not good for online transaction processing (OLTP) environment

The most common sizes of the I/O requests are 4, 8, and 16 KB for transaction-oriented processing applications and 16–32 KB for sequential data processing. It is therefore recommended to use the relevant block size in Iometer for similar applications that customers wish to test.

However, for applications such as backup, video, and audio streaming using SCSI streaming commands (SSC), 32–64 KB I/O sizes are the more common block size. As streaming media technology expands, different demands and needs will arise, and system tuning will change to address these needs with varied block size. Other tunable parameters may exist for enhancing performance for these applications. However, so far, from block size perspective, 32–64 KB is still the common best practice.

By default, most tape backup and recovery applications use either 32 KB or 64 KB I/O sizes as the default tape block size setting. In addition to the data compression consideration, the optimal block transfer size of each discrete tape drive model varies. Some drives work best with 64 KB transfers, and some deliver optimal performance at block sizes of 32 KB. Larger block sizes do not necessarily result in improved throughput. The architectural foundation of each particular drive model dictates the optimal transfer size. For example, for DLT drives, a 64 KB I/O size is the optimal buffer size for the best performance.

Using Microsoft Exchange Server as a sample application, different I/O block sizes are used for different functions. Without going into the architectural details of each application, an exchange database, for example, actually consists of two files: the properties store .EDB and the streaming store .STM. These files have different access characteristics, depending upon the type of clients that will be supported. For example, the streaming store .STM file typically uses a 32–64 KB block size.

Table 3 summarizes which Microsoft Exchange I/O simulation profiles can be used within Iometer.

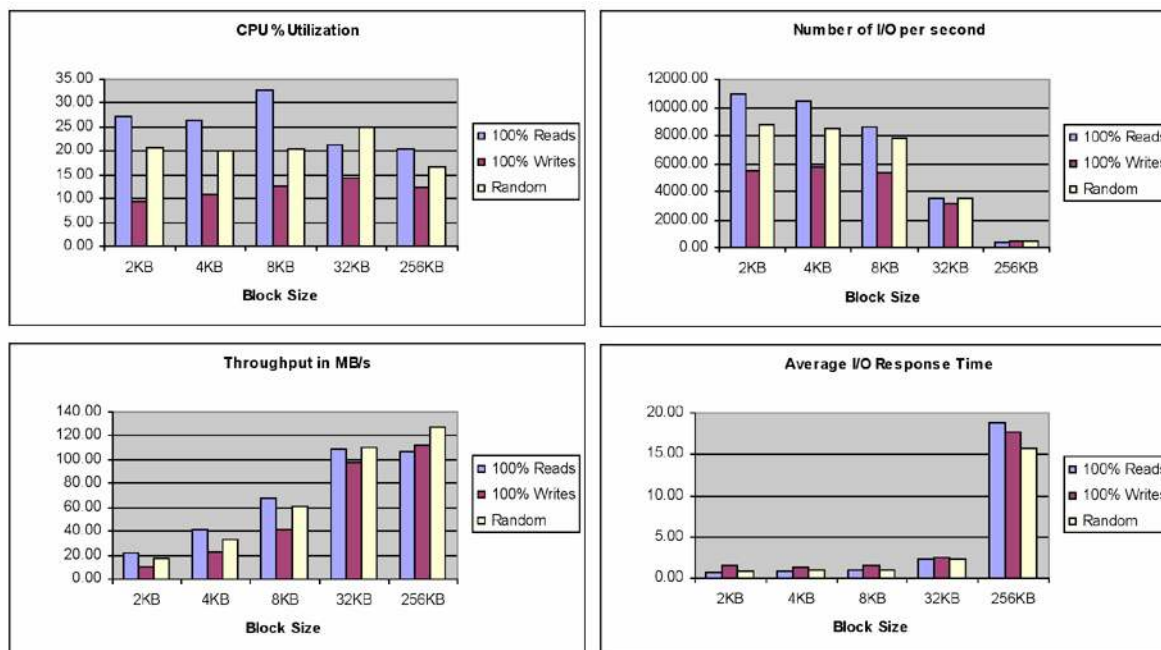
Table 3

Using Microsoft Exchange I/O Simulation Profiles with Iometer

Database Store	Logical block size (KB)	Characteristic of I/O
.EBD	4	Random (50/50 R/W)
.STM	32-64	Random (50/50 R/W)
.LOG	4	Sequential (100 W)

Based on a configured iSCSI test, Figure 1 shows four different Iometer performance test results based on block sizes ranging from 2 KB to 256 KB. Keep in mind that these numbers are provided only as reference points for a single host with iSCSI access to show CPU, I/O operations per second (IOPS), throughput, and latency. With different hardware, different versions of the iSCSI host driver, and different disks, the performance numbers could be different.

Figure 1



Iometer Performance Test Results

SUMMARY

When running performance tests using Iometer from a single host, multiple factors will affect performance. In most cases, the host is usually the bottleneck. If the host has a slow processor and insufficient memory, test performance will be affected. The fact that iSCSI uses processing power from host CPU for the iSCSI stack could potentially affect performance as well. One can compensate for this by using multiple hosts to generate more iSCSI aggregate I/O or by using a single initiator with higher CPU and memory resources. The other major performance factor will be the block size of the I/O hitting the disks or tape. When determining the optimal block size for application performance, keep in mind that the block size range mentioned in this paper must be used only as a guideline. If customers have specific applications that they wish to use Iometer to run the simulation tests on, modifications must be made in block size and other I/O parameters to make the I/O test closely resemble the application I/O. Finally, the number of disks and the number of outstanding I/Os will also affect the performance. Generally speaking, the more disks and the higher number of outstanding I/Os that are configured, the higher the aggregate I/O performance.

The most important thing to consider is to ensure the chosen I/O parameters for the test closely resemble the I/O patterns of the target production applications for iSCSI. Running a 1 MB I/O size profile, for example, does not resemble any realistic application and will only serve to consume time because 99 percent of today's applications will never use I/O sizes above 64 KB.

APPENDIX A

Clariion CX600 Configuration

Depending on which storage array is used, host LUN masking might need to be configured on the storage array. In the example, EMC's Clariion CX600 is used, and host LUN masking needs to be configured. With Access Logix being used on the Clariion, creation of RAID groups, storage groups and LUNs will be needed beforehand so that the iSCSI client can access the LUNs. How the LUNs are configured through the RAID group will affect the performance. The more physical disks the LUNs are mapped to, with the better the performance. The Clariion CX

models support RAID 0, RAID 5, and RAID 1+0. Consult EMC engineers for exact configuration of LUNs on the Clariion. On the EMC Clariion, after the iSCSI client has logged into the storage port, the Clariion needs to register that host to that specific storage port.

APPENDIX B

Below are the CLI configurations on the Cisco MDS switch and how to accomplish each task:

1. Enable iSCSI on Cisco MDS switch

- Switch# config terminal <enter>
- Switch(config)# iscsi enable <enter>
- Switch(config)# end <enter>

2. Configure Gigabit Ethernet interface

- Switch# config terminal <enter>
- Switch(config) interface gigabitethernet 3/3 <enter>
- Switch(config-if)# ip address 10.10.10.1 255.255.255.0 <enter>
- Switch(config-if)# iscsi authentication none <enter>
- Switch(config-if)# no shutdown <enter>
- Switch(config-if)# end <enter>

3. Configure iSCSI interface

- Switch# config terminal <enter>
- Switch(config)#interface iscsi 3/3 <enter>
- Switch(config-if)#mode store-and-forward <enter>
- Switch(config-if)#no shutdown <enter>
- Switch(config-if)# end <enter>

4. Create iSCSI initiator

- Switch# config terminal <enter>
- Switch(config)#iscsi initiator ip-address 10.10.10.87 <enter>
- Switch(config-(iscsi-init))# vsan 1 <enter>
- Switch(config-(iscsi-init))# end <enter>

5. Auto import of FC targets

- Switch# config terminal <enter>
- Switch(config)# iscsi import target fc <enter>
- Switch(config)# end <enter>

6. Create ZoneSet and Zone

- Switch#config terminal <enter>
- Switch(config)# zoneset name iSCSI vsan 1 <enter>
- Switch(config-zoneset)# zone name iSCSI_host1 <enter>
- Switch(config-zoneset-zone)# member pwnn 50:06:01:68:10:60:14:f5 <enter>
- Switch(config-zoneset-zone)# member ip-address 10.10.10.87 <enter>
- Switch(config-zoneset-zone)# end

7. Activate ZoneSet

- Switch#config terminal <enter>
- Switch(config)#zoneset activate name iSCSI vsan 1 <enter>



Corporate Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters

Cisco Systems International
BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: 31 0 20 357 1000
Fax: 31 0 20 357 1100

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters

Cisco Systems, Inc.
168 Robinson Road
#28-01 Capital Tower
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the Cisco Web site at www.cisco.com/go/offices.

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia • Cyprus
Czech Republic • Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland
Israel • Italy • Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland
Portugal • Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden
Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

Copyright © 2004 Cisco Systems, Inc. All rights reserved. Cisco, Cisco Systems, the Cisco Systems logo, Catalyst, and EtherChannel are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or Web site are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0402R)

203234_ETMG_RD_07.04

Printed in the USA

