

Realtime
publishers

The Shortcut Guide[™] To



**Exchange
Server 2007
Storage Systems**

sponsored by



Jim McBee

Chapter 4: Best Practices for iSCSI Storage Systems	89
Volume (LUN) Management.....	90
Volume Provisioning	90
LUN Allocation Per Database	92
Thin Provisioning.....	92
Volume Naming.....	93
Persistent Targets	95
Documenting the LUN Assignments	96
Optimizing the Partition's Starting Sector and Formatting the Disk.....	98
Disk Labeling.....	101
Operations and Performance Monitoring of iSCSI Volumes	104
Managing an Exchange Server that Uses an iSCSI SAN	104
Monitoring Disk Performance on iSCSI Volumes	105
Improving Performance and Storage Availability	109
Basic Steps	109
Multi-Path I/O.....	110
Backup Recommendations.....	116
Summary	118

Copyright Statement

© 2007 Realtimerepublishers.com, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtimerepublishers.com, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtimerepublishers.com, Inc or its web site sponsors. In no event shall Realtimerepublishers.com, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtimerepublishers.com and the Realtimerepublishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtimerepublishers.com, please contact us via e-mail at info@realtimerepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library. All leading technology guides from Realtimepublishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 4: Best Practices for iSCSI Storage Systems

The previous chapters of this guide discussed the appropriate path for properly sizing your disks for both storage capacity as well as I/O capacity. This path has to include not only the maximum theoretical capacity you estimate you might need but also the maximum I/O load that you think you will need to support. This ensures that you run out of neither storage nor I/O capacity during the planned lifetime of your storage system.

Chapter 3 covered some of the basics of using the iSCSI initiator client on Windows 2003 to connect to a LUN that has been provided on an iSCSI target. That chapter just touched on the basics of implementing the iSCSI initiator.

This chapter will cover some of the ways you can avoid common mistakes that new system administrators make when they implement an iSCSI SAN as well as best practices you can follow to ensure that you avoid problems and run your iSCSI SAN system as efficiently as possible. Topics in this chapter include:

- iSCSI volume (LUN) management
- Operations and performance monitoring of iSCSI volumes
- Ways to improve performance and storage availability
- Backup recommendations


There are a couple of salient points with which I would like to start this chapter. First and foremost is to ensure that you start with up-to-date software both on the Windows and SAN side. For example, I have seen periods of time where Microsoft has released new versions of the iSCSI initiator every 2 or 3 months in order to provide improvements in performance or new features. SAN vendors often release new versions of their SAN software as well as supporting software for Windows. Ensure compatibility between the SAN operating system (OS), the Windows iSCSI initiator, and the SAN vendor's additional software that runs under Windows. Often vendors will post a matrix of minimum versions of software required; ensure that you follow this matrix when configuring (or updating) systems.

☞ Ensure that all software on the Windows side and the SAN is interoperable.

For most Exchange and network administrators, iSCSI is a relatively new technology. Most administrators that I work with when helping to deploy Exchange on a new iSCSI SAN have never worked with either an iSCSI or a fibre channel SAN. Testing and developing a pilot or lab system first are important parts of a new deployment. Doing so helps you to learn and become familiar with managing an iSCSI system and avoid corrupting the data found on the SAN.

The second important part of an iSCSI deployment is to generate good documentation. This documentation should allow you to quickly determine how your iSCSI SAN is configured and which LUNs are configured for what purpose.

Finally, your SAN vendor is the last (and arguably the most important) piece. Choose a vendor with which you are able to establish a good dialog and for whom you are given specific technical or account contacts. The versions of Windows, Exchange Server, and the iSCSI initiator that you want to use may change; your iSCSI SAN vendor may also have firmware/software that you need to consider. As new versions of the software are released, you want to have someone that you can refer questions regarding compatibility and interoperability. A good technical representative from your SAN vendor can keep you on the path of a consistent and supported configuration.

 Establish a good relationship with your SAN vendor.

Volume (LUN) Management

The most logical place to start when talking about iSCSI SAN management and best practices is to start with the LUNs themselves. Logically, if you correctly configure and connect your LUNs between the iSCSI initiator and the iSCSI target systems, you are well on your way to a solid configuration.

Volume Provisioning

Each SAN vendor and implementation of iSCSI SAN use a slightly different terminology to refer to how they collect and allocate the physical disks on the SAN. For example, Network Appliance refers to a collection of disk segments as an aggregate; from that aggregate, individual LUNs are carved and assigned to iSCSI initiators. LeftHand Networks combines an entire SAN group into a single logical storage container. The Open Filer software that I used in Chapter 3 for examples combines usable disk space into volume groups. Figure 4.1 shows a screen shot from the OpenFiler management interface in which I am combining three physical disks (well, actually, VMware disks) into a single volume group.

Create a new volume group

Volume group name

VG1

Select physical volumes to add

<input checked="" type="checkbox"/>	/dev/sdb1	2.00 GB
<input checked="" type="checkbox"/>	/dev/sdc1	2.00 GB
<input checked="" type="checkbox"/>	/dev/sdd1	2.00 GB

Add volume group

Figure 4.1: Creating a volume group that consists of three physical disks.

SAN software that allows you to individually manage the physical disk drives in the system and combine them into logical groups gives you the ability to size not only the disk storage capacity but also the I/O capacity. If you know that you need a specific IOPS requirement for a LUN, you could combine enough physical disks to give you the necessary IOPS capacity. Although this may seem appealing from a techie perspective, it might be more “tuning” than most of us actually want to do.

The approach that SAN vendors take is to aggregate all the available disk drives in the entire SAN into a single unit. The fault tolerance is handled behind the scenes by the SAN OS. By combining all the physical disks for the entire SAN into a single volume group or aggregate, you get the full I/O capacity of all the disks in the entire SAN. Consult with your SAN vendor to ensure that your total IOPS capacity is at least what you expect to use.

When you create LUNs, make sure that you define enough disk space to meet or exceed your requirements. Most SANs on the market today will allow you to expand the size of an existing LUN “on the fly.” To add disk space to an existing LUN, you must have the additional disk space available on the SAN.

LUN Allocation Per Database

For Exchange Server 2003, you should allocate two LUNs per storage group up to a maximum of eight LUNs. One of the LUNs should be sized for transaction logs and one should be sized for all the databases in that storage group. If a storage group has more than one database, those databases should be on the same LUN as the other databases in the storage group.

For Exchange Server 2007, each storage group should contain one database. Each storage group should be allocated a total of two LUNs; one LUN will hold the storage group's transaction logs and checkpoint file and the other will hold the database for the storage group. It is a best practice to scale upward on storage groups by creating more storage groups; if you need 15 databases, you should create 30 LUNs (15 LUNs for the storage group's transaction files and 15 LUNs for the databases).


By placing one database in each storage group, you are able to configure and use local continuous replication or cluster continuous replication. If you choose to employ volume shadow copy or snapshot backups, the granularity of the backup and restore is at the database level, provided the storage group has only a single database in it.

Thin Provisioning

Some SAN vendors support a concept known as *thin provisioning*. This can help make better use of your unallocated disk space pool. Let's say, for example, that you have an Exchange database that you know will grow to 200GB in size. Today however, your Exchange database is only 50GB in size. With traditional volume provisioning methods, you could create a 200GB volume today, and have 150GB in unused space sitting idle.

If the SAN supports *volume expansion*, you could create a 50GB volume based on your current needs, and manually expand that volume when your Exchange database grows. If your SAN vendor or OS does not support volume expansion, you would have to create a new, larger volume, and move your data to that new volume.

With thin provisioning, you can create a volume that is 200GB in size, but only uses 50GB in actual physical space in the SAN. In this case, the SAN presents 200GB to the Exchange Server, but only reserves the space on the SAN that is actually needed—50GB. As that 50GB increases, due to adding more users or larger mailboxes, the SAN will expand the amount of space it has reserved and is using for this volume, without the application knowing that it is happening. As far as the application is concerned, it has had the full 200GB capacity from the time the volume was created. The SAN only uses the amount of actual space that is required from the Exchange Server. There is nothing to reconfigure on the Exchange Server.

 Thin provisioning enables the SAN to dynamically increase the size of a LUN within a specified set of growth parameters.

The benefit of thin provisioning is that it allows systems administrators to more effectively use their current resources. There is no need to waste space by configuring the SAN for requirements that are projected, and there is no need to reconfigure the hosts or SAN as volume use increases.

You should have some type of monitoring or alert system in place to let you know that a LUN has been increased in size so that you can monitor the situation and determine whether administrative action should be taken, such as archiving users' mailboxes. Although database growth is usually part of normal operations, it should be monitored to ensure that it is normal growth and not due to some unexpected event.

Some vendors support tools that can shrink the size of a LUN if you have allocated too much space. However, even if your SAN solution does not provide this feature, if you need to shrink the size of a LUN back to its original size, you can move all the data off the LUN, remove the LUN completely from the SAN, and then recreate it at the desired size.

Volume Naming

Documentation is one of the most important tasks when you configure a system. The better your documentation, the more quickly you can resolve problems or update a configuration later. As part of the documentation process, I consider a very important step the process of creating volume names that are intuitive and easy to follow. If the volume name indicates exactly what a volume or LUN is intended to be used for, there is less chance of mistakes in the future.

Sure, volume names like *volume20201xy* are good for job security, but I personally will forget what I meant that volume to be used for within about 10 seconds of creating it. Take the example in Figure 4.2; in this example, I am carving an iSCSI LUN out of the volume group (*volumegroup1*) I previously created. This LUN is going to be the home of the database SFOEX01-MB01; this is mailbox database 01 on the server SFOEX01.

Block storage statistics for volume group "volumegroup1"

Total Space	Used Space	Free Space
4128768 bytes (4032 MB)	0 bytes (0 MB)	2064384 bytes (4032 MB)

Create a volume in "volumegroup1"


Volume Name (must be specified like a UNIX filename without its path)	<input type="text" value="SFOEX01-MB01"/>
Volume Description	<input type="text" value="SFO Exchange MB01"/>
Required Space (MB)	<input type="text" value="700"/> 
Filesystem type	<input type="text" value="iSCSI"/>
<input type="button" value="Create"/>	

Figure 4.2: Use a naming convention that is intuitive and helpful.

A good naming standard that is intuitive and standardized will prove immediately helpful for the SAN operator. Figure 4.3 shows the OpenFiler's existing volumes list. Scanning through this list, it is pretty obvious which volumes are intended for which purpose to the SAN administrator.

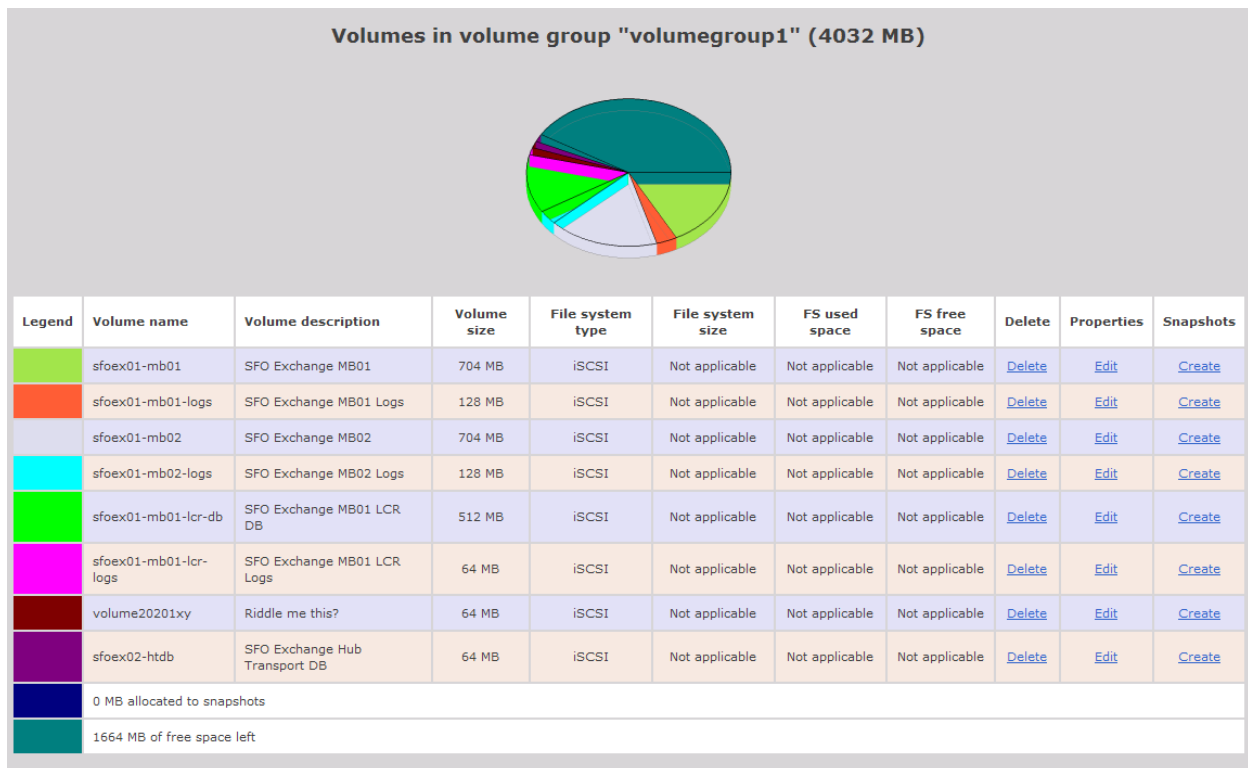


Figure 4.3: A good naming standard will pay off.

Figure 4.3 shows you just how easy it is to identify the purpose of each volume or LUN that is created on this particular iSCSI SAN. In an environment in which the person that manages the SAN is different than the person that manages Exchange, this is particularly helpful. However, this will pay off in other areas too, such as when you are configuring the Windows iSCSI Initiator Client to connect to this particular iSCSI target. Figure 4.4 shows the iSCSI Initiator Control Panel applet after connecting to an iSCSI SAN. Notice how easy it is to identify, based on the LUN name, the intention of each of the LUNs.

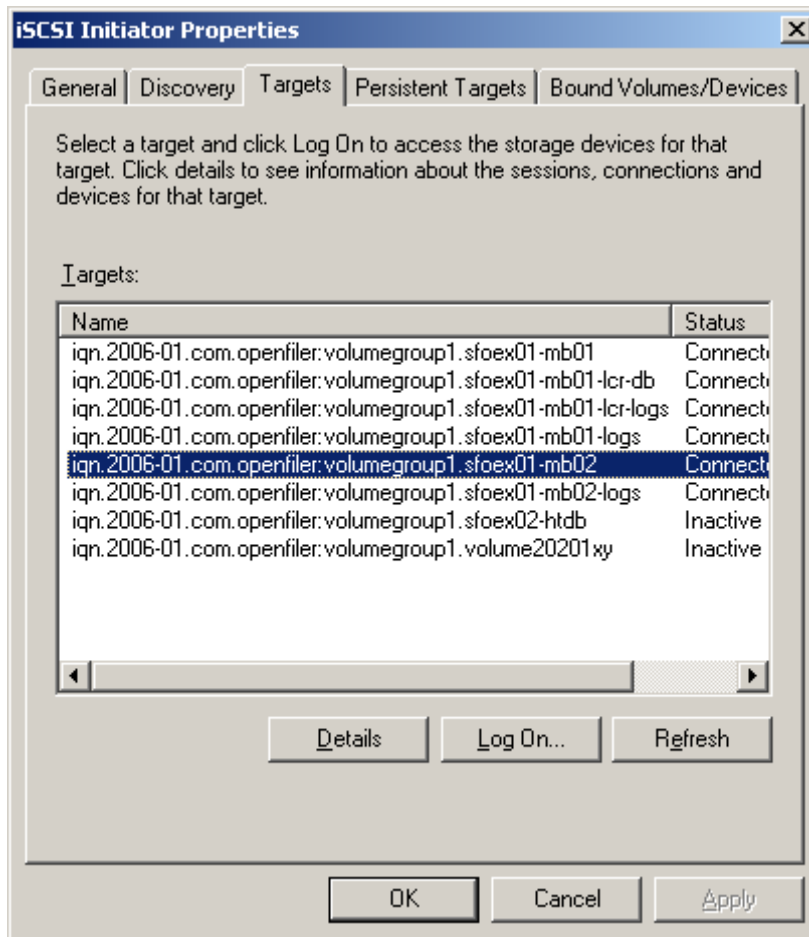


Figure 4.4: Viewing available LUN names in the iSCSI Initiator Control Panel applet.

Persistent Targets

When configuring the Windows iSCSI Initiator Client to connect to iSCSI target LUNs, always make sure that the target LUN is configured as persistent. You can configure a persistent target when you initially log on to the target LUN. Figure 4.5 shows the Log On to Target dialog box. To make a LUN persistent, select the *Automatically restore this connection when the system boots* check box.

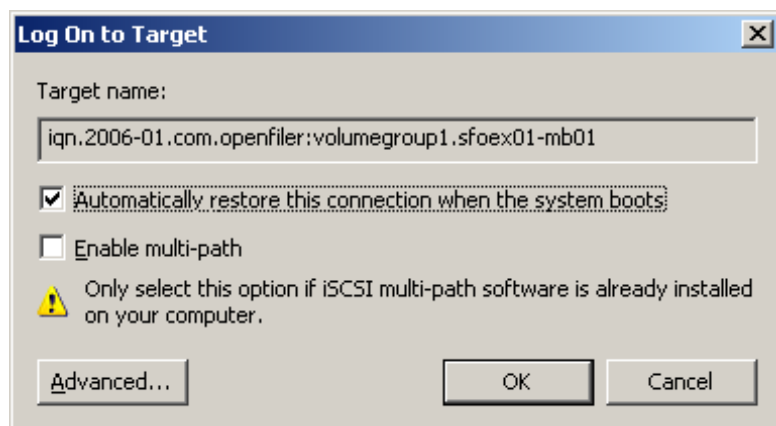



Figure 4.5: Enabling persistence for an iSCSI LUN.

If you forget to enable persistence for a target LUN, you will find yourself having to reconnect the LUNs each time the Windows system restarts. This is probably not acceptable for application servers such as Exchange Server or SQL Server.

Documenting the LUN Assignments

For each server that is using iSCSI LUNs, you should create documentation that will help the administrator cross reference the LUN with the drive letter or the mount point. One way you could do so is when you create the LUN to assign each LUN a slightly different size; they would then be noticeably different when you format them. That is probably not a terribly good practice, though. Instead, you want to cross reference these devices by using the SCSI address of the LUN. Doing so will require that you use both the iSCSI Initiator Control Panel applet and the Windows Disk Administrator.

 Many third-party providers of iSCSI SAN software may also provide software for Windows that makes the process of mapping LUNs both different and easier.

To get this information from the iSCSI Initiator, use the iSCSI Initiator Control Panel, navigate to the Targets property page, select each target, click Details, select the Devices property page, and click Advanced. Doing so will display the General property page for the Device Details. The left side of Figure 4.6 shows information for Target ID 3 (device number 4). In this example, the target is `iqn.2006-01.com.openfiler:volumegroup1.sfoex01-mb01-lcr-db`, which represents the SAN volume called `sfoex01-mb01.lcr.db`.

The right side of Figure 4.6 shows the properties of the disk when viewed in the Windows Disk Management console.

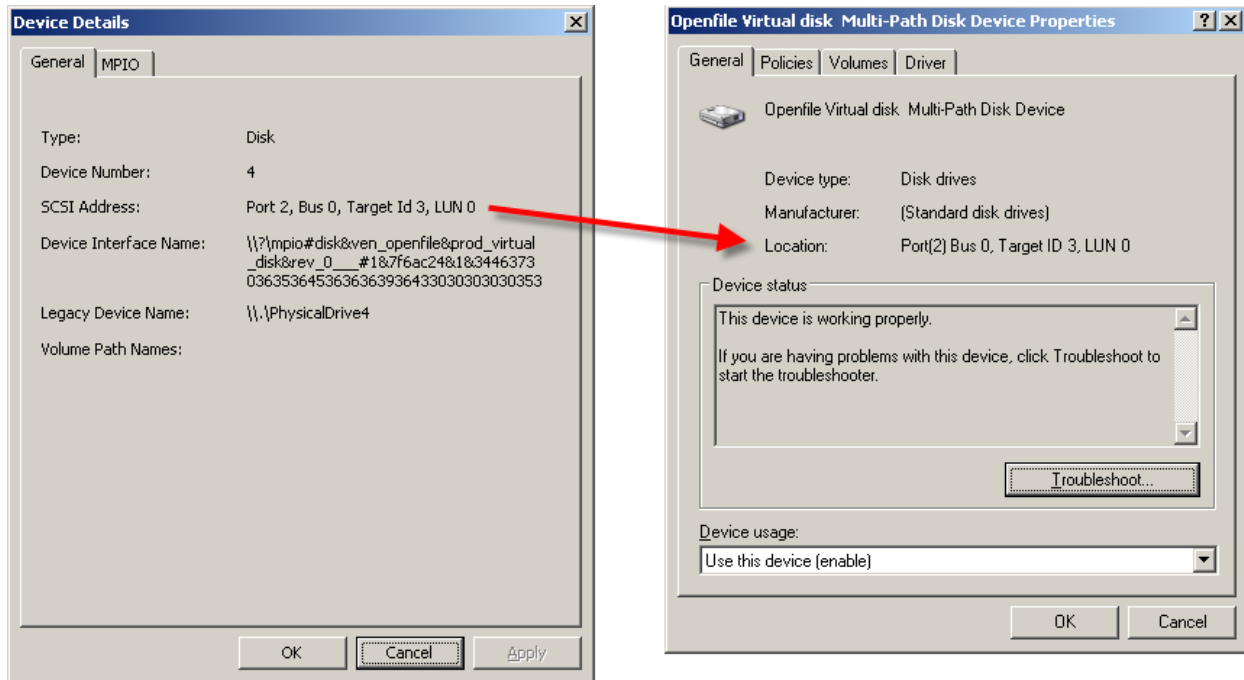


Figure 4.6: Cross-referencing the iSCSI target to a disk number.

In the Disk Administrator program, this disk was listed as Disk 4; this is shown in Figure 4.7. All this information needs to be documented so that you can modify or rebuild this server in the event of disaster or system updates.

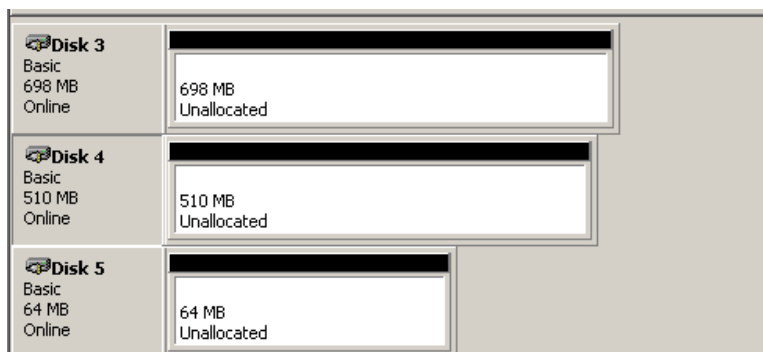


Figure 4.7: Viewing disk numbers in the Windows Disk Management console.


From the information we record from the iSCSI and target properties, we want to build a table that lets us see the drive letter or mount points for each of the targets. Table 4.1 shows what this might look like for this server.

Disk number	Drive Letter or Mount Point	SCSI ID	SAN volume	Purpose/Notes
0	C:\	Local	N/A	OS and binaries
1	F:\	Port(2) Bus 0, Target ID 0	sfoex01-mb01	Mailbox database SFOEX01-MB01
2	I:\	Port(2) Bus 0, Target ID 1	sfoex01-mb01-logs	Transaction logs for mailbox database SFOEX01-MB01
3	G:\	Port(2) Bus 0, Target ID 2	sfoex01-mb02	Mailbox database SFOEX01-MB02
4	H:\	Port(2) Bus 0, Target ID 3	sfoex01-mb01-lcr-db	LCR location for SFOEX01-MB01
5	K:\	Port(2) Bus 0, Target ID 4	sfoex01-mb01-lcr-logs	LCR location for SFOEX01-MB01
6	J:\	Port(2) Bus 0, Target ID 5	sfoex01-mb02-logs	Transaction logs for mailbox database SFOEX01-MB02


Table 4.1: Server SFOEX01 disk volumes.

Optimizing the Partition's Starting Sector and Formatting the Disk

When you use the Windows Disk Management console program to create a partition on a disk, the partition wizard creates the partition starting at the 64th sector of the disk; this is because Microsoft allows as many as 63 hidden sectors on a disk for non-Windows uses (such as a third-party vendor utility.) Windows can format a disk in block sizes that are 4KB, 8KB, 16KB, 32KB, or 64KB. If the starting disk block spans two sectors (which is what will happen if Windows starts the first block on sector 64 rather than 65), the disk block will span two sectors on the disk. This results in inefficient read and write performance. By some estimates, this reduces the I/O capacity of the disk by 20%.

 Do not use DiskPart.exe's partition management functions on a disk that is already formatted and has data on it.

To alleviate this potential performance problem, always configure disk partitions that will be used by Exchange by using the Windows 2003 diskpart.exe utility. With this utility, you can manually specify the starting sector of the disk. Check with your vendor to make sure this is a good idea on the particular SAN you own, but it is almost always a good practice.

 Properly aligning the starting sector of a partition can noticeably improve performance.

Although I covered this with an example in Chapter 3, I think it is a good idea to go over it again using the SAN example I have been following so far in this chapter. Previously, in Table 4.1, we documented the disks and their intended purposes. If we open a command prompt, run the `diskpart.exe` utility, and use the List Disk command, we see these disks again:

```
Microsoft DiskPart version 5.2.3790.1830
Copyright (C) 1999-2001 Microsoft Corporation.
On computer: SFOEX01
```

```
DISKPART> list disk
```

```

Disk ###  Status      Size      Free      Dyn Gpt
-----  -
Disk 0    Online      16 GB     8033 KB
Disk 1    Online      698 MB    698 MB
Disk 2    Online      128 MB    128 MB
Disk 3    Online      698 MB    698 MB
Disk 4    Online      510 MB    0 B
Disk 5    Online      64 MB     64 MB
Disk 6    Online      128 MB    128 MB

```

Suppose I want to create a partition for Disk 1. I would use the Select Disk command and the Detail Disk command to verify information about the disk:

```
DISKPART> select disk 1
```

```
Disk 1 is now the selected disk.
```

```
DISKPART> detail disk
```

```
Openfile Virtual disk Multi-Path Disk Device
Disk ID: D013300A
Type   : iSCSI
Bus    : 0
Target : 0
LUN ID : 0
```

```
There are no volumes.
```

This helps to confirm that the disk truly does not have a volume currently created, and it allows me to confirm that it is the correct Target ID (in this case Target ID 0) that I want to partition.

Next, I want to create the partition, so I need to specify the starting sector. I use the Create Partition Primary Align=X command to create the partition. I will specify an alignment value in KB. For Exchange 2003, you can use an alignment of either 32 or 64, but for volumes that will be used by Exchange 2007, use 64. Here is the result of that command and the detail disk command again:

```
DISKPART> create partition primary align=64
```

```
DiskPart succeeded in creating the specified partition.
```

```
DISKPART> detail disk
```

```
Openfile Virtual disk Multi-Path Disk Device
```

```
Disk ID: D013300A
```

```
Type : iSCSI
```

```
Bus : 0
```

```
Target : 0
```

```
LUN ID : 0
```

```
Volume ### Ltr Label Fs Type Size Status Info
-----
* Volume 3 Partition 698 MB Healthy
```

Finally, the last thing you need to do is to assign this drive a drive letter. You can either do so from the DiskPart utility (using the Assign Letter command) or you can assign the drive letters using the Disk Management console.



You can find good supplemental information about DiskPart.exe (and its predecessor DiskPar.exe) on the Exchange team blog at <http://msexchangeteam.com/archive/2005/08/10/408950.aspx>.

Once all the partitions are created with the correct starting offset, I can now format the partition. It is probably easiest to go back to the Disk Management console to format each of the newly created partitions. For Windows 2003 disks, format the partition with a 4K allocation unit size, but for Exchange 2007 servers, format the partition using a 64K allocation unit size (see Figure 4.8).

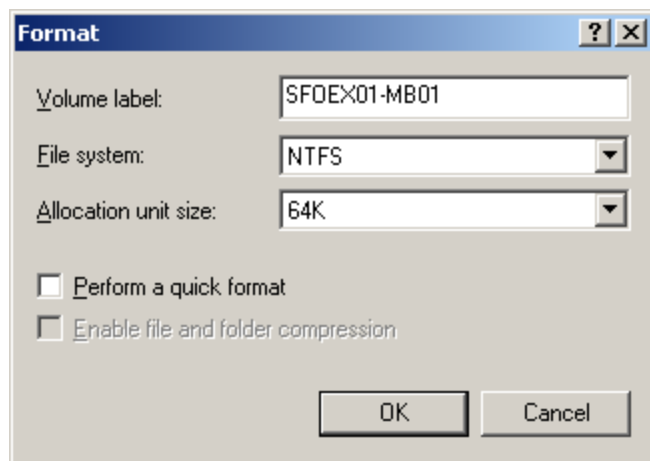
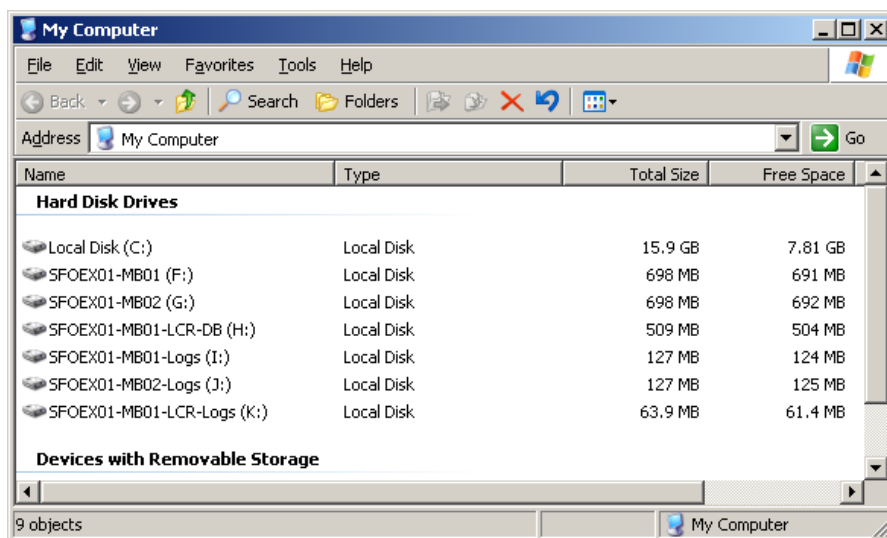


Figure 4.8: Selecting the allocation unit size.

Disk Labeling

Just as important as labeling the LUNs/volumes on the SAN is ensuring that LUNs are correctly identified by the Windows administrator. Once Windows initializes the disk, a unique disk signature is applied to the disk, but that does not help the administrator much. Making sure that the drives are labeled correctly does not seem like much of a big deal at first, but remember with iSCSI LUNs that the drive could accidentally be assigned to a different server entirely! I have seen situations with SANs and with clusters where a drive got “lost” and accidentally assigned to another server that was using the SAN. Believe me, if this ever happens, you will want to quickly be able to determine the purpose of the LUN and where it really belongs!

The first step is to assign the partition a unique volume label when you format it. I did this previously in Figure 4.8. By doing so, I can easily identify the purpose of the disk either from Windows Explorer or the Disk Management console (as shown in Figure 4.9.)



Volume	Layout	Type	File System	Status
(C:)	Partition	Basic	NTFS	Healthy (System)
SFOEX01-MB01 (F:)	Partition	Basic	NTFS	Healthy
SFOEX01-MB01-LCR-DB (H:)	Partition	Basic	NTFS	Healthy
SFOEX01-MB01-LCR-Logs (K:)	Partition	Basic	NTFS	Healthy
SFOEX01-MB01-Logs (I:)	Partition	Basic	NTFS	Healthy (Active)
SFOEX01-MB02 (G:)	Partition	Basic	NTFS	Healthy
SFOEX01-MB02-Logs (J:)	Partition	Basic	NTFS	Healthy

Disk 3 Basic 698 MB Online	SFOEX01-MB02 (G:) 698 MB NTFS Healthy
Disk 4 Basic 510 MB	SFOEX01-MB01-LCR-DB (H:) 510 MB NTFS

Figure 4.9: Disk drive labeling in action.

You might think we are through with the documenting and labeling, but we have one more step. The disk label will not be preserved if a LUN is mounted by another Windows server, so we want to take a very basic step to provide some additional insurance with respect to labeling. We will take a very simple approach to this; we will create a text file in the root of each disk. The name of the file will identify the server, purpose, SAN volume name, and drive letter (or mount point) that should be assigned to the label. So, for example, Disk 1 in Figure 4.9 would have a text file in the root of the F:\ drive that would be named as follows:

```
SFOEX01-ExchangeMailboxes-SFOEX01-MB01-F-Drive.txt
```

Alternatively, another naming scheme for this file name would be something that identified the server name, drive letter, SAN name, and the SAN volume. Here is an example:

```
SFOEX01_Drive-G_SFOSAN01_SFOEX01_MB01.txt
```

Regardless of what naming scheme you use, pick something that will be informative for your organization. I would record the following information in the text file:

```
Home server name:          SFOEX01
SAN Volume:                sfoex01-mb01
SAN Name:                  SFOSAN01
Server drive letter:      F:\
```

This may seem like overkill, but I have seen situations in which a LUN was assigned by accident to a different server and the administrator spent valuable time trying to figure out what data was on the disk and where it belonged. This technique is also quite valuable when setting up clustered servers, as the text file in the root of the disk will easily help identify the intention of the volume. Figure 4.10 shows how easy it is to identify the disk's intention by just looking at its root directory using Windows Explorer.

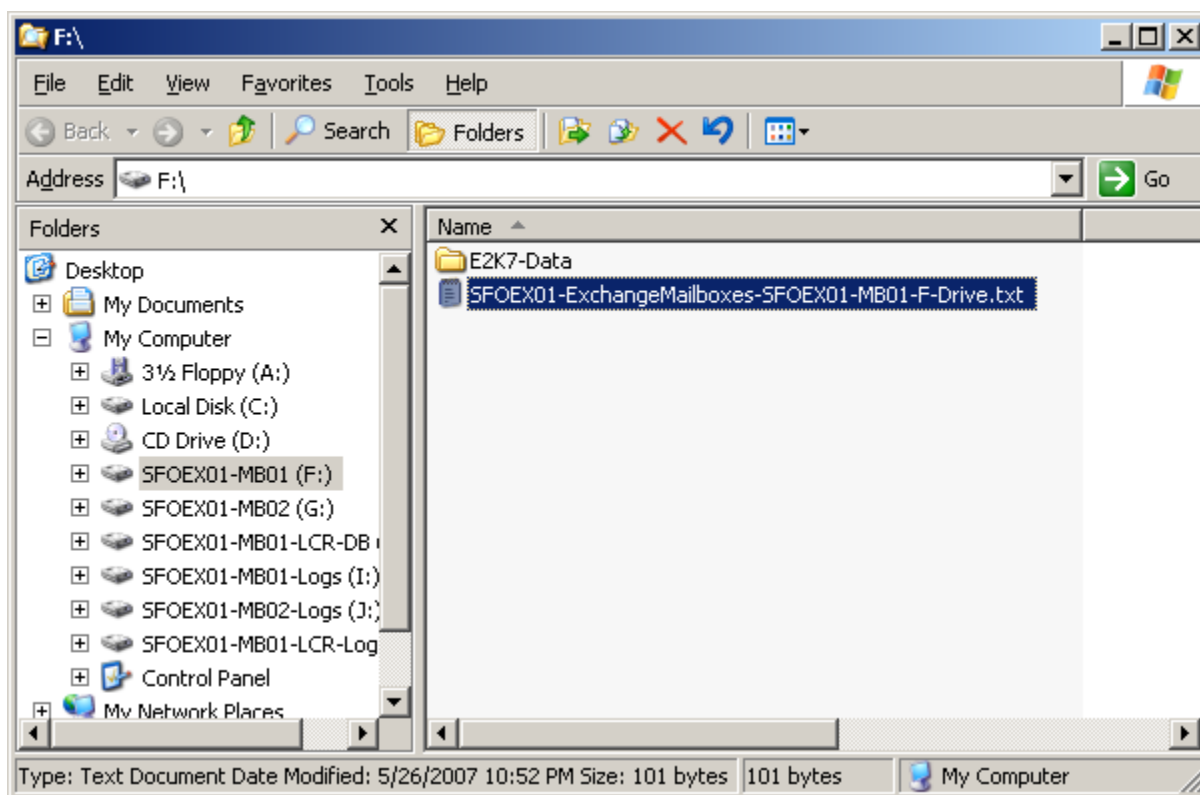


Figure 4.10: Viewing the text file in the root of the disk drive.

Things such as properly labeling the disk volume and putting helpful disk “label” files in the root of a volume are easy to do and you can do them very quickly. These quick tasks will more than pay for themselves in terms of time and worry if you ever have a situation that requires quick knowledge of volumes and the intended use of the volumes.

Operations and Performance Monitoring of iSCSI Volumes

This section explores a few key things you should keep in mind with respect to daily operations and performing health checks and monitoring of your disk volumes. Tools such as Microsoft Operations Manager can help you in monitoring the health of your Exchange Server when using an iSCSI SAN, but even the built-in tools such as Performance Monitor are helpful once you know what to look for.

Managing an Exchange Server that Uses an iSCSI SAN

From the Exchange Server's perspective, there is not any difference between using locally attached (direct attached) disk storage and using volumes that are connected to the server via an iSCSI or fiber channel SAN. The disks are presented to the applications by the OS as local disk drives.

However, there is a bit of difference when it comes to actual operations and management of the Exchange Server. Most of the differences are a result of the fact that the actual disk storage is not only physically separate from the physical Exchange Server but also there is a separate OS that maintains the disk storage. If you have ever managed a server that has an externally attached disk subsystem, you will begin to have a feel for what it is like. However, externally attached disk subsystems still connect to the server via a local SCSI channel rather than being run by an independent OS.

First and foremost, develop simple, concise documentation for startups, shutdowns, and reboots. This should include power-on and power-off sequences and times to wait between tasks. Post these clearly so that all operators and administrators are familiar with them. Next, never perform maintenance (even a reboot) on an iSCSI SAN without first dismounting all the Exchange databases and stopping the Microsoft Exchange Information Store service. Ideally, you should always shut down the Exchange Server prior to doing any work on the SAN. Next, never assign the same iSCSI target LUN to two servers unless you are using Microsoft Cluster Services. If the cluster service is not managing access to the LUNs, one server initiator will corrupt data the other initiator is accessing. Finally, consult with your SAN vendor to determine whether there are additional operating or management requirements for your iSCSI SAN.

Another good tip is to implement some type of monitoring solution that will let you know when your disk space on any particular server volume is running low. One of the most frequent causes of downtime in any Exchange organization is when the server runs out of disk space on a transaction log or database disk. Having these files on a SAN with lots of storage or even thin provisioned volumes does not necessarily mean you will not exceed the amount of space you have allocated.

Monitoring Disk Performance on iSCSI Volumes

Regardless of how carefully you calculate the necessary I/O capacity and plan for that capacity on the SAN, it is still possible that your users are just going to use the system in ways that you did not anticipate. For this reason, periodic health checks are important; checking the health of the disks is especially important because that is most often the place that is affected first by an overzealous user community or unexpected usage growth. Some monitoring systems will report on potential disk problems.

When using the Windows System Monitor console, there are a few key counters to watch when monitoring your disk performance:

- Physical Disk/Disk Transfers/sec indicates the number of I/O operations (read and write) that are taking place for a specific physical disk or LUN. The maximum value for a LUN will depend entirely on its I/O capacity.
- Physical Disk/Avg. Disk sec/Transfer indicates how long it takes to either read or write data to the disk. On average, this value should remain below .02 or 20 milliseconds per transfer.
- MExchangeIS/RPC Averaged Latency indicates an average of the last 1024 RPC packets and is displayed in milliseconds. This value should always be below 50ms. In my experience, high values of this counter indicate network problems, but they can also indicate that the Exchange Server is not servicing requests quickly enough.

Performance tuning and performance monitoring are as much of an art as they are a science. Sure, you can monitor for specific thresholds and raise alerts if that threshold is exceeded, but there are a number of factors you should take into consideration:

- Monitor over a period of typically heavier email system usage. For many businesses, this is Monday morning between 8:00AM and 11:00AM. However, don't take one day's worth of monitoring as being typical. Monitor for several days during typical periods of usage.
- Don't sweat the spikes of activity if they are only occasional. You are interested in averages over a typical usage period.
- Even regular spikes of activity where the Physical Disk object's Avg. Disk sec/Transfer counter is sustaining a value of 40 or 60 milliseconds may not be cause for concern. Always check to see whether there is a noticeable delay in message delivery or when a user opens a message.
- Don't take just one day's monitoring as being typical. Monitor during the same set of hours for a few days and then repeat the following week.

Let's take a few examples of situations in which a server is under an I/O load that is too high for its current capacity. Figure 4.11 shows the Physical Disk/Disk Transfers/sec counter for a server's E:\ drive. In this example, there are four Exchange databases on this disk; there are approximately 900 users using these four databases. The E:\ drive disk is actually a RAID 5 array with 10 physical disks.

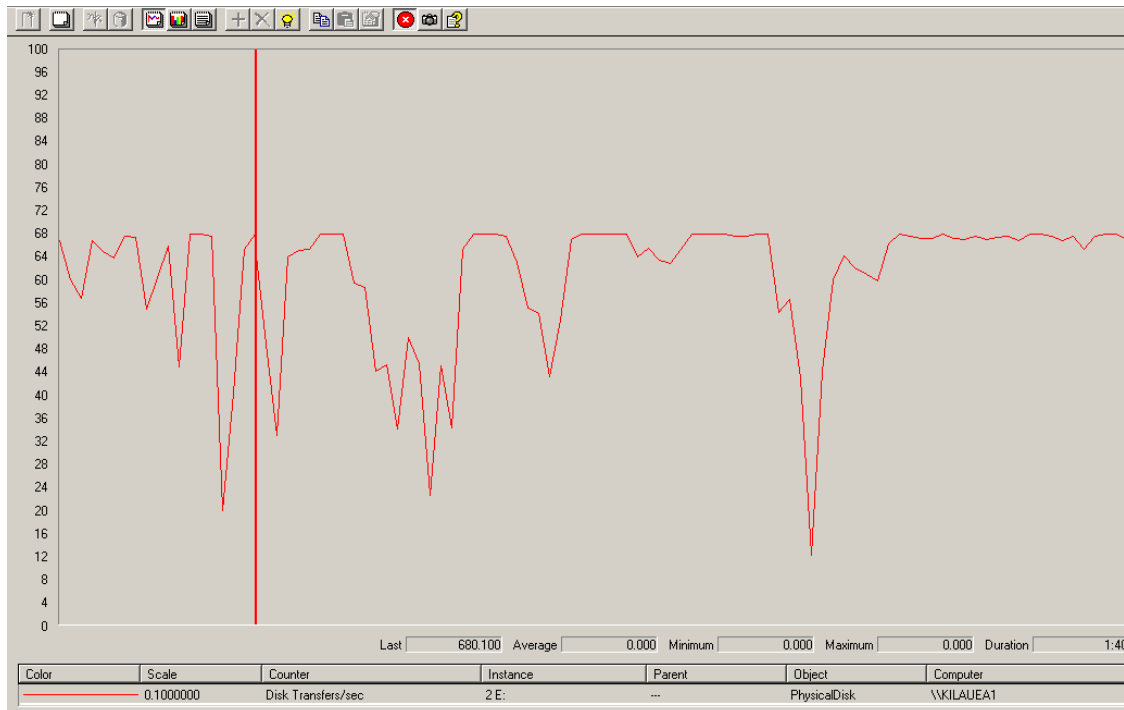


Figure 4.11: Monitoring the Disk Transfers/sec counter.

What we are actually seeing in Figure 4.11 is the disk operating at maximum I/O capacity during this short monitoring period. This logical disk is capable of sustaining no more than about 680 IOPS or disk transfers per second. Although the fact that this disk is sustaining up to 680 IOPS per second is not really critical, what is more important is that we are reading from or writing to the disk in a timely fashion. This is where the Physical Disk's Avg. Disk sec/Transfer counter comes in handy. Figure 4.12 shows the F:\ drive of an Exchange Server that is supporting approximately 430 simultaneous heavy users and has two databases on the F:\ drive.



Figure 4.12: Monitoring the Avg. Disk sec/Transfers counter.

Though Figure 4.12 represents a very small portion of time (approximately 100 seconds) for this Exchange Server, during this time, the average transfer took 50 milliseconds. (Note that the scale for this graph is 1000.) The peak during this time was just over 300 milliseconds. Ideally, we would use the Performance Monitor and Alerts console's logging feature and log the disk activity over a period of typical usage, such as from 8:00AM until 11:00AM or whatever is typical for this particular organization. Once I have recorded the log file, I will use the System Monitor tool to display and analyze the results I have recorded to the log file.

Microsoft has an additional tool for Exchange administrators if you are trying to get to the bottom of performance problems in general. This is the Exchange Performance Troubleshooter that is now built-in to the Microsoft Exchange Troubleshooting Assistant. Exchange 2003 users can download this tool from <http://technet.microsoft.com/en-us/exchange/2007/bb288482.aspx>. If you are running Exchange 2007, you will find this tool in the Tools work center of the Exchange Management Console. One of the options when you launch the Performance Troubleshooter is to check for updates; I recommend you do so each time you run this tool so that you are working with the latest configuration and best practices.

Figure 4.13 shows the Microsoft Exchange Troubleshooting Assistant when running in Exchange 2007. You will find the tool and the reports it produces to be very similar to the Exchange Best Practices Analyzer (<http://www.exbpa.com>). The report includes information about disk configuration, data and log file locations, processor health, and RPC performance.

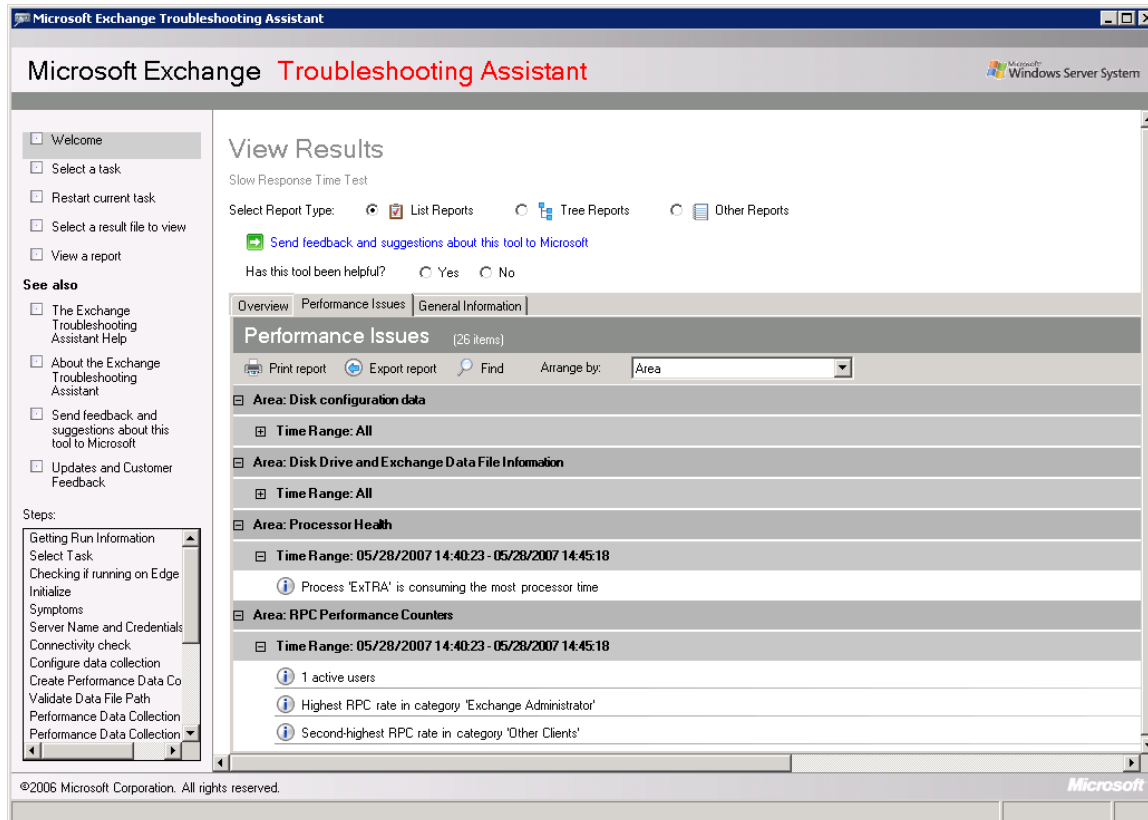


Figure 4.13: Viewing an Exchange Troubleshooting Assistant performance report.

Improving Performance and Storage Availability

There are a number of things that you can do both during the design phase of your iSCSI SCAN planning and in the deployment and operation that will help to improve the availability, reliability, and performance of the system.

Basic Steps

Let's start with some of the more basic improvements and quick fixes you can do for your iSCSI system. Some of these basic steps and procedures may seem obvious, but when working with a newer technology or something you have never worked with before, it is always best to assume as little as possible:

- Check with your vendor for optimization and deployment tips. I know I am sounding like a broken record with regards to checking with your vendor, but any place you can get shared knowledge on implementing a new type of system will be helpful.
- Use v2 or later of the Microsoft iSCSI Initiator for improved performance.
- Use Gigabit Ethernet adapters. For server-to-SAN connections, you can use a cross-over cable, but if you have more than one server connecting to a single SAN, you must use a Gigabit Ethernet switch. 100MB networks are fine if you are trying to learn the technology, but the performance will be unacceptable.
- Always use a dedicated Gigabit Ethernet connection for server-to-SAN connections. Although you can share the server-to-SAN network with other types of traffic, you should not do this in production.
- Do not use NIC teaming for connections that will support iSCSI targets.
- If you require more throughput than you can get with Gigabit Ethernet cards, consider using iSCSI host bus adapters (HBAs).
- Ensure that all LUNs that will be used at server start time (such as the LUNs used by Exchange databases and transaction logs) are configured as persistent connections.
- If the Ethernet switches are configurable, use jumbo frames and flow control to ensure the most reliable performance.
- For Exchange 2007 Mailbox servers, follow the Microsoft RAM recommendations for heavy users. This is at least 2GB of RAM plus 5MB of RAM for each mailbox.
- For Exchange 2003 Mailbox servers, put 4GB of RAM in the server and follow the recommendations found in Microsoft article 815372 "How to Optimize Memory Usage in Exchange Server 2003."

Multi-Path I/O

In any environment that is going to require significant throughput between the initiator and the target, you should consider the use of multi-path I/O. Multi-path I/O will improve the performance of the connection between the initiator and the target and will provide you fault-tolerance. The Microsoft iSCSI Initiator and most SANs on the market support multi-path I/O. Table 4.2 shows the expected initiator-to-target throughput when using the latest version of the Microsoft initiator and Gigabit Ethernet. (The values are in megabytes per second, not megabits per second.)

Storage NICs	Estimated Data Throughput
1 NIC	92 MB/second
2 NICs	185 MB/second
3 NICs	241 MB/second

Table 4.2: Estimated throughput with multiple NICs.

You should plan your multi-path I/O into your design so that you have all the hardware and software necessary to implement it. This is especially important when ordering hardware so that you have sufficient network adapters in both the servers and the iSCSI SANs as well as having the necessary Gigabit Ethernet switches.

Figure 4.14 shows two possible designs. The design on the left uses a dedicated Gigabit network for iSCSI SAN communication and can be configured to use the public network connection as a failover path. This will provide some redundancy in case the dedicated SAN network fails, but it should not be used for any type of load balancing of the data communication between the initiators and the target. This design is simple to implement as most server-class hardware that is shipped today includes two Gigabit Ethernet adapters.

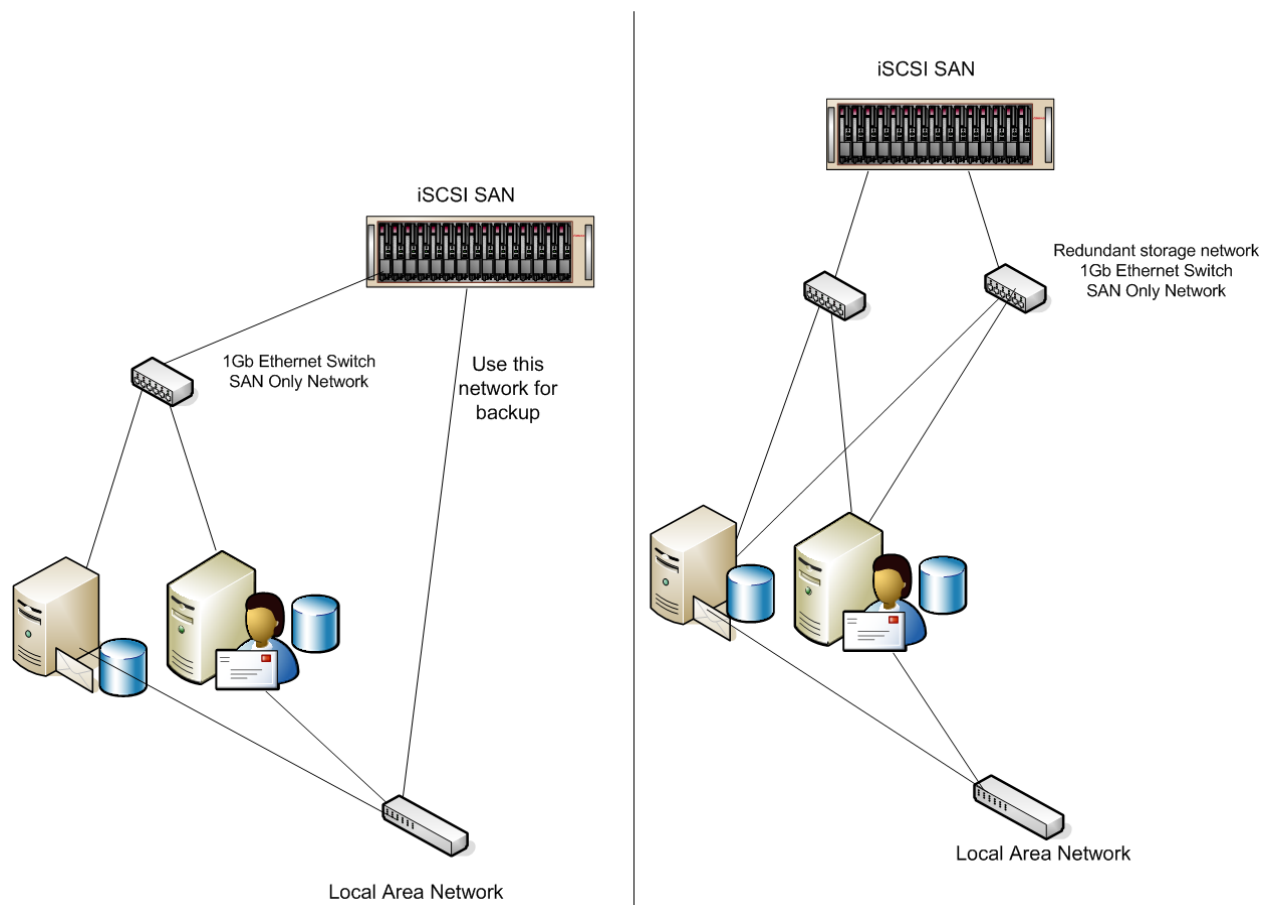



Figure 4.14: Possible multi-path connections to iSCSI SAN.

The right side of the diagram is a bit more complex and requires three NICs for the iSCSI SAN hardware and each of the servers. In this design, each server has two connections to two different dedicated SAN networks. You can then configure multi-path I/O (MPIO) settings to use a round-robin algorithm to distribute the data load between the two networks evenly. There are a number of different load-balancing algorithms available when you configure the MPIO options.

When you select the Load Balance Policy drop-down list, you will see a couple of options available for load balancing or failover:

- Fail Over Only uses a single active path for data transfer between the initiator and the target. All other paths are used as standby paths. If more than one standby path is available, the initiator will employ a round-robin approach to find a path that will work. This is the solution you would use if you wanted to use a dedicated SAN network all the time but fail over to the public network if the dedicated SAN connection fails.
- Round Robin uses a load-balancing technique to evenly distribute the I/O between the initiator and the target. This is the best solution for most multi-path I/O solutions.
- Round Robin With Subset uses the round-robin policy on active paths but will use other paths designated as standby if there is a failure on the active paths.

- Least Queue Depth attempts to compensate for unevenly distributed I/O by using less loaded I/O paths.
- Weighted Paths allows the administrator to assign a weighting value for each path. If you select this value, the Edit button on the MPIO property page will let you assign a weighting value to each path. A higher value means that particular path has a lower priority.
- Least Blocks routes I/O requests to the processing path with the least number of pending I/O blocks.

 Please take the multi-path I/O example in this chapter as generic information. Your actual implementation may be different based on your own design, requirements, or SAN vendor's recommendations.

In previous examples, I have stuck to a single connection to the iSCSI SAN and I have not configured the initiators for multi-path I/O. I want to go through an example and explain some of the additional terminology. First and foremost, though, during the installation of the iSCSI Initiator software, you must make sure that you have included the Microsoft MPIO Multipathing Support for iSCSI (see Figure 4.15.) If you have neglected to do so, you can go back and run the installation again, but a reboot may be required. If you have already mapped LUNs and are using them, make sure that you stop any services (such as the Exchange information store) that are using the LUNs before you attempt to install additional iSCSI initiator options.

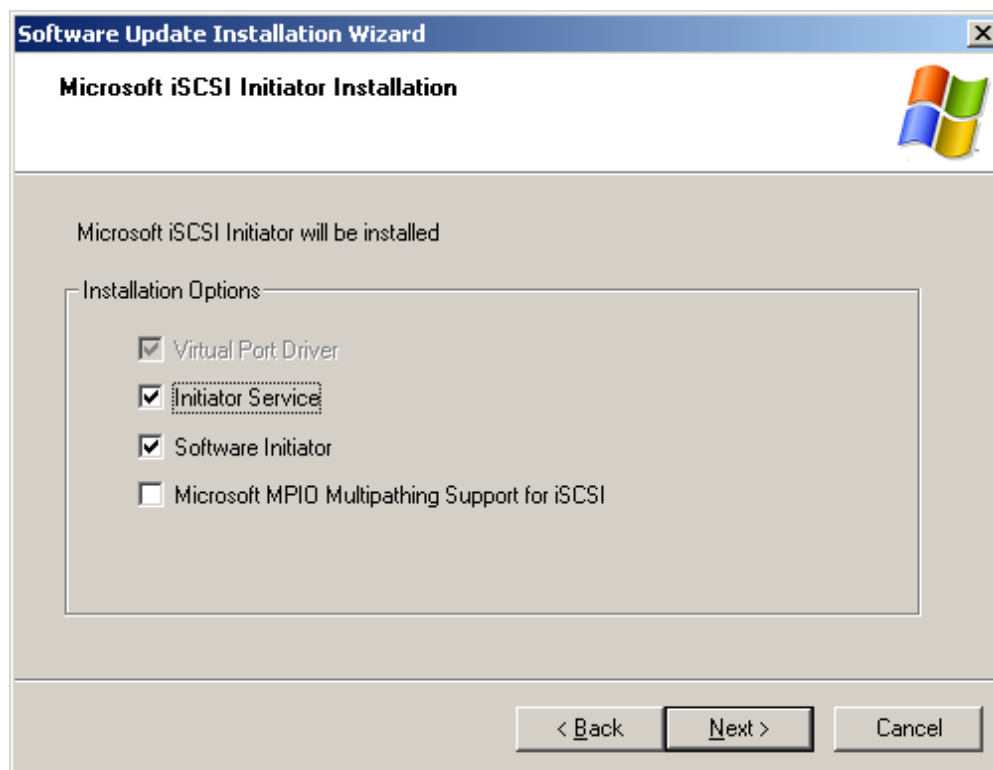


Figure 4.15: Installing the iSCSI Initiator client and the multi-path option.

Next, when you configure the iSCSI Initiator properties, on the Discovery property page, make sure that you include the IP addresses for both connections to the iSCSI SAN. Figure 4.16 shows the Discovery property page with two separate target portals.

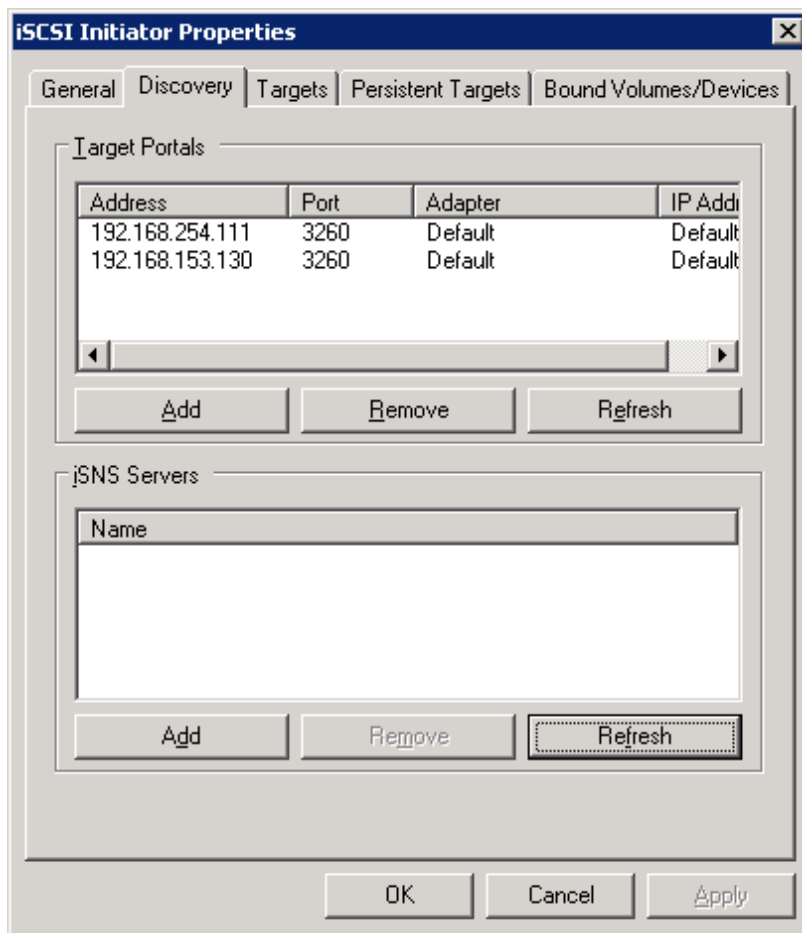


Figure 4.16: Discovery property page for the iSCSI Initiator Control Panel applet.

Next, we have to configure each of the iSCSI targets. Switch to the Targets property page in the iSCSI Initiator Control Panel applet. For each iSCSI target on the Targets property page, click Log On and ensure that the Enable Multi-Path check box is selected on the Log On to Target dialog box (see Figure 4.17).

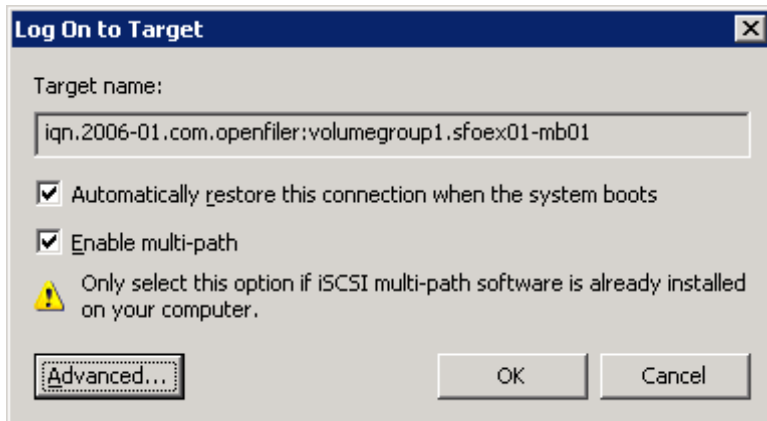


Figure 4.17: Configuring each target for multi-path I/O.

Next, we have to configure the multi-path I/O pieces for each target LUN. Highlight each target and click Details. On the Target Properties page, make sure the Sessions tab is selected, and click Connections to confirm that there are two sessions for this particular target. The left side of Figure 4.18 shows that this particular target has two sessions.

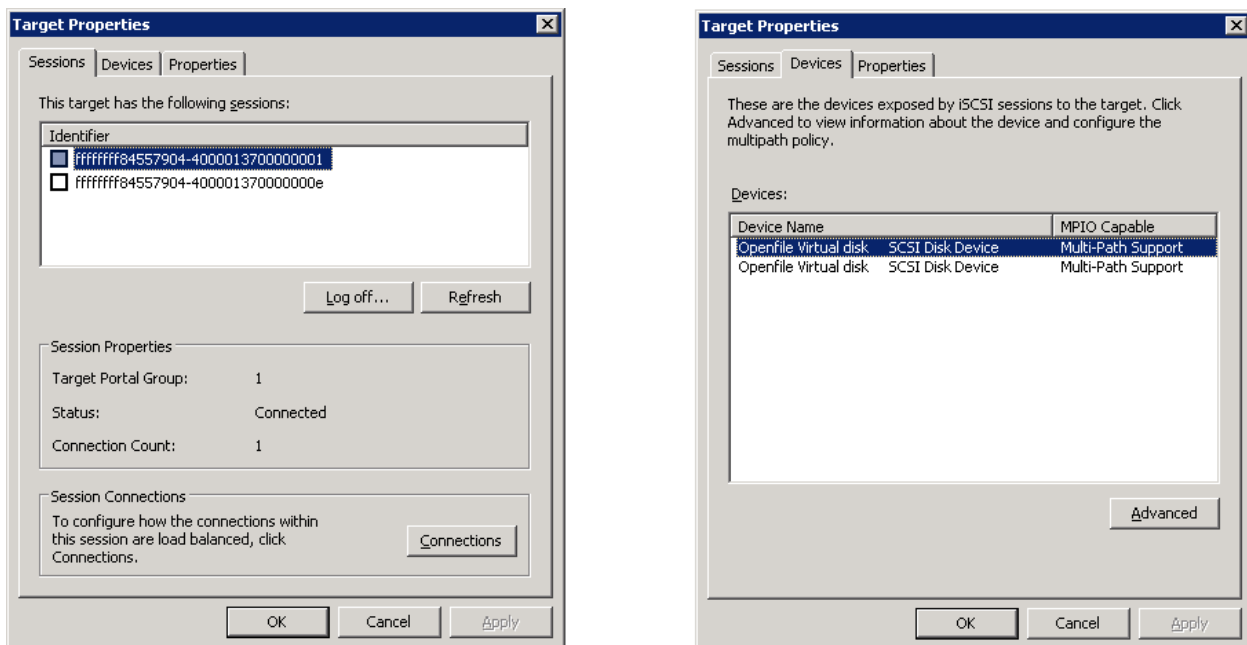


Figure 4.18: Viewing the sessions for a particular target.

On the Devices property page, you should confirm that there are two devices associated with this particular target; the right side of Figure 4.18 shows the Devices property page.

Next, we need to specify the multi-path load balancing property. For the design that we are doing, we want a round-robin policy that will distribute the load evenly between the two connections. If we were going to use the public network, we would use the failover only policy. These are configured on the Devices property page (shown on the right side of Figure 4.18) of the target by clicking Advanced. On the Device Details properties page, select the MPIO property page (shown in Figure 4.19). You can select the load-balance policy by selecting the Load Balance Policy drop-down list.

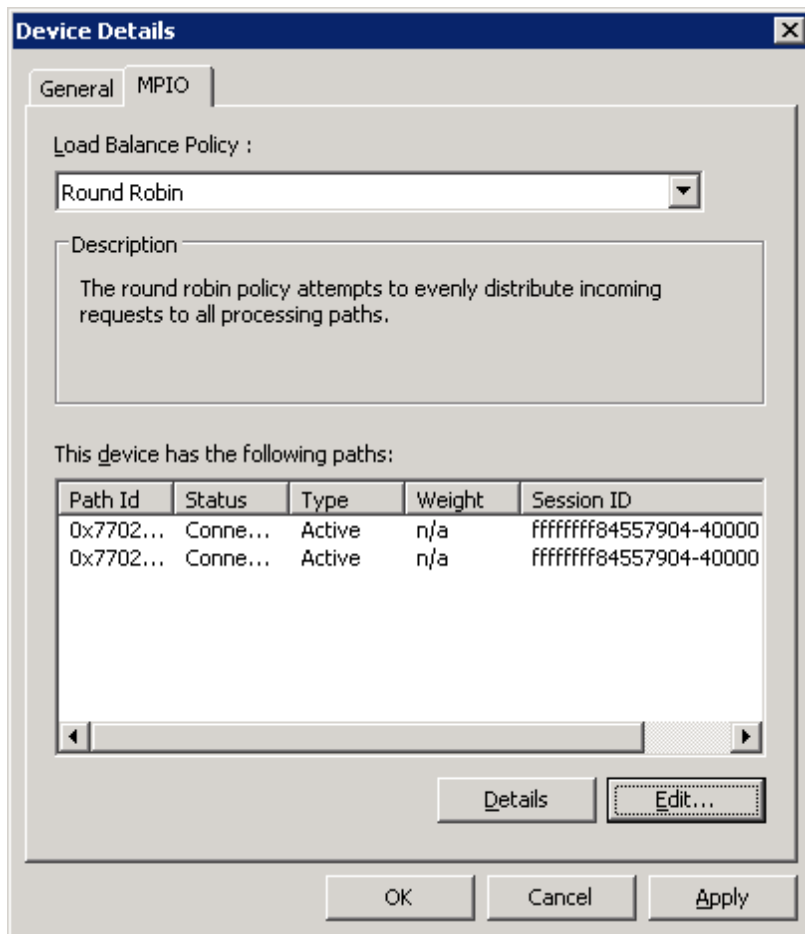


Figure 4.19: Selecting an MPIO load-balance policy.

When you have selected the load-balancing property, click OK until you have closed all the properties for this particular target. When finished configuring all targets, close the iSCSI Initiator Control Panel applet and reboot the server in order for multi-path I/O to take effect.

Backup Recommendations

Both Exchange Server 2007 and SANs introduce new approaches to backing up data. The time it takes to back up Exchange data to streaming tape media, or correction, more importantly the time it takes to restore from tape, is a major design consideration when it comes to figuring out the maximum amount of data you can allow each user to have. By decreasing restore times, you can increase the amount of data each user is allowed to have because you can restore it more quickly if necessary.

A traditional “online” Exchange backup does not back up the database files directly but rather uses the Exchange backup APIs to back up the database one page at a time. As each page is backed up, the page’s checksum and page pointers are verified. If a damaged page is found, the Exchange online backup reports the error and halts. The intention is that if an online backup is completed successfully, you know that not only is your data backed up but also that the data file is not corrupted. Offline backups (backing up the data file with the database dismounted) and “old school” snapshot backups do not do this by default.

Snapshot technology is certainly nothing new; storage and OS vendors have been doing this for a long time. Exchange, though, presents a special challenge: the Exchange data file is never truly consistent unless it is dismounted because there is always the possibility that some of the data is in memory and has not yet been committed to the data file. This did not stop vendors from coming up with proprietary mechanisms for taking Exchange snapshots and verifying the integrity of the snapshot.

With the release of Windows 2003 and Exchange 2003, Microsoft introduced a snapshot API that allows vendors to use the Exchange APIs as part of the snapshot backup process. Any backup solution that you choose that will work with your iSCSI SAN should use the Microsoft VSS and Exchange 2003 or Exchange 2007 VSS requestor. If you are planning to use snapshot backups, consider the following tips:

- For Exchange 2003, scale outward with storage groups before filling a storage group with mailbox or public folder stores. Each storage group should have two LUNs; one for the transaction logs and one for that storage group’s database.
- For Exchange 2007, scale outward with storage groups. If you need eight databases, you should have eight storage groups. Each storage group’s database should have a dedicated LUN and each storage group’s transaction logs should have a LUN. If you configure your LUNs this way, the granularity of your snapshot backups and restores will be per database.
- Allow additional space on the LUN for the snapshot backups. Depending on the snapshot technology, this may be between 110% to 150% of the size of the database being backed up. This will also vary based on the number of days that you want to keep for a recovery window. I usually recommend a 14-day recovery window.
- Make sure you have a plan for long-term retention of your Exchange data. Your snapshot recovery Window will be only for a limited period of time (such as 14 days) and is really intended only for disaster recovery. If you have requirements to restore historical Exchange data further back than your snapshot window, you will need some alternative media or mechanism available.

See Microsoft's TechCenter article "Best Practices for Using Volume Shadow Copy Service with Exchange Server 2003" at <http://technet.microsoft.com/en-us/library/aa996004.aspx> for more information about using VSS backups and Exchange Server.

Taking a snapshot (or restoring a snapshot) is quick (often almost instantaneous for 100GB of data or more). Some Exchange backup solutions that take advantage of snapshot technologies will perform a verification of the database using the ESEUTIL utility. If you run database verifications, this can increase the I/O during the backup. This might interfere with normal operations on the Exchange Server if you use the Exchange Server to perform the verification. If the Exchange Server is in use by end users during the snapshots and the backup software needs to perform a verification of the snapshot database, try to offload the verification to a different server. I have used Exchange front-end servers and domain controllers for this task.

Let's extend the discussion of snapshot backups to Exchange Server 2007; Exchange 2007 introduces the new continuous replication features Local Continuous Replication (LCR) and Cluster Continuous Replication (CCR). Both of these technologies allow you to keep a replicated and nearly up-to-date copy of your Exchange data either on the local Exchange Server (in the case of LCR) or on a passive clustered node (in the case of CCR). Figure 4.20 shows LCR illustrated. As the transaction logs are filled up and closed on the E:\ drive, they are then copied to the G:\ drive, verified, and then committed to backup copies of the databases on the H:\ drive.

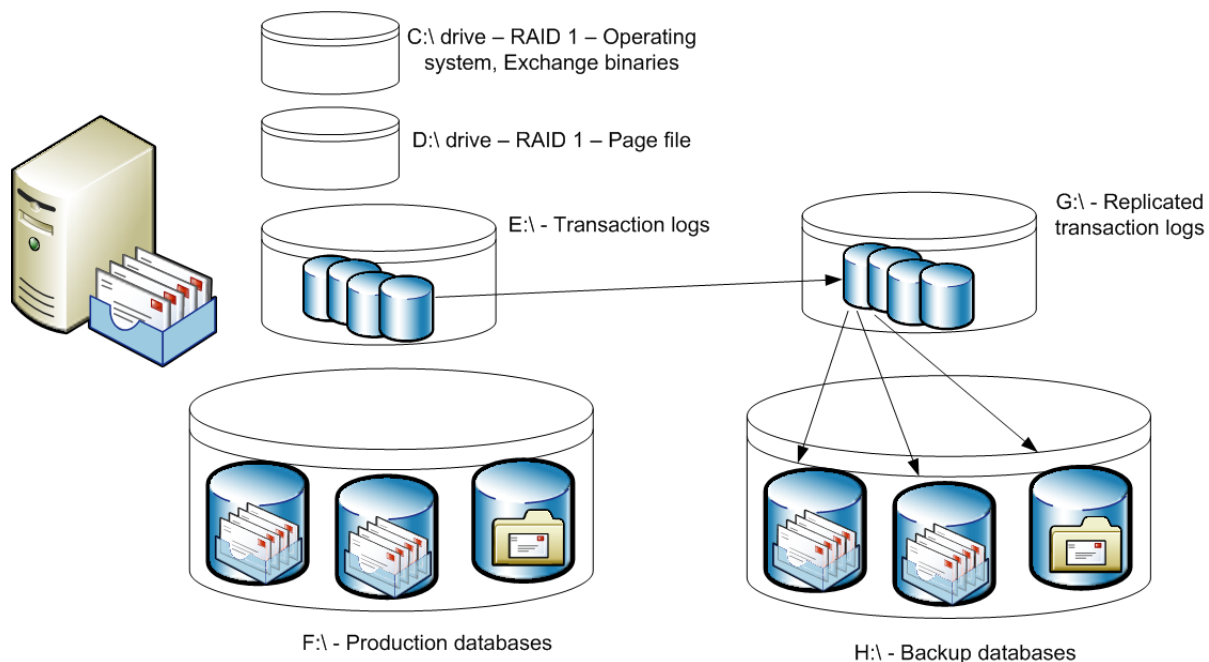


Figure 4.20: LCR for Exchange 2007.

By themselves, the replication technologies in Exchange 2007 will allow you to increase the maximum size of the database. Microsoft recommends a maximum Exchange 2007 database size of 100GB without implementing continuous replication and 200GB if you implement continuous replication. With either of these replication technologies, you always have a backup copy of the data ready to bring on line. With LCR, you can easily bring online a replicated copy of a failed database and with CCR you can bring online the passive clustered node.

You might be wondering how this relates to iSCSI SANs, though. In Figure 4.20, there is nothing stopping you from putting the E:\, F:\, G:\, and H:\ drives on an iSCSI SAN. Then, rather than taking snapshots of the production data found on the E: and H: drives, you would snapshot the replicated copies. This would significantly reduce the I/O required to take snapshot backups.

Summary

In closing, I wanted to re-iterate a few important points and tips that I have covered in this chapter that will help you to implement a better iSCSI SAN environment that provides you with the utmost in reliable storage service:

- Always put more physical disk space in your SAN than you expect to consume. Assign your aggregates or volume groups at least 20% more disk space than you initially expect to carve out of them in LUN space.
- Use Gigabit Ethernet cards and switches for your SAN network topology. Doing so will ensure the best possible performance. For even greater throughput between the iSCSI initiator and target, implement multi-path I/O.
- When performing upgrades of the Microsoft Windows iSCSI Initiator Client or any other SAN-based software, first perform a full backup of all Exchange databases and then stop the Microsoft Exchange Information Store service and disable it. This will ensure that all Exchange databases are mounted during a software upgrade.
- When initializing iSCSI LUNs using the Windows Disk Management console, do not convert the disks to dynamic disks. iSCSI target LUNs should remain basic disks.
- During startups and shutdowns, always remember that the Exchange Server is shut down first and started up last. Give the SAN plenty of time to start and make the iSCSI LUNs available.
- Use Windows 2003 DiskPart.exe to properly align the starting partition on all partitions that contain Exchange data.
- Develop good procedures for keeping your SAN and Exchange servers documented. Follow these procedures rigidly.
- Periodically evaluate the current versions of SAN and Windows software available to you. If there are updates that you should apply, carefully plan the upgrade to ensure compatibility between initiator and target through the upgrade.

- When sizing LUNs and volumes on your SAN, do not forget to take into consideration disk space required for snapshots. This number will vary based on the snapshot technology, the number of changes each day, and the number of days of historical information you want to keep. For an Exchange database volume, you may need to estimate snapshot capacity 110 to 150% of the size of the databases on that volume.
- If you are using tape-to-disk backups and you have more than one iSCSI SAN, consider implementing a separate path to a second SAN for storing backup files. This will reduce the load on the “data” SAN during backups and allow you to separate backups on to a separate system.

I hope this chapter and guide have served as a useful introduction to the concepts of sizing Exchange properly for both storage and I/O capacity and for implementing Exchange storage on an iSCSI SAN. As you begin to investigate how iSCSI SANs can work for your organization, you will find that they are easily and cost effectively implemented.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.