Chapter 9

Storage

When designing a Virtual Infrastructure environment, one of the single most important things to consider and plan for is the storage backend. There are several options available that range from local storage, Fibre Channel and iSCSI. The first thing to think about is where you store and run your virtual machines. VMware's VMFS file system is specially designed for the purpose of storing and running virtual machines.

Virtual Machine File System

VMware developed its own high performance cluster file system called VMware Virtual Machine File System or VMFS. VMFS provides a file system which has been optimized for storage virtualization for virtual machines through the use of distributed locking. A virtual machine stored on a VMFS partition always appears to the virtual machine as a mounted SCSI disk. The virtual disk or *.vmdk file hides the physical storage layer from the virtual machine's operating system. VMFS versions 1 and 2 were flat file systems, and typically only housed .vmdk files. The VMFS 3 file system now allows for a directory structure. As a result, VMFS 3 file systems can contain all of the configuration and disk files for a given virtual machine. The VMFS file system is one of the things that set VMware so far ahead of its competitors. Conventional file systems will allow one server to have a read/write access or lock to a given file at any given time. VMware's VMFS is a file system which will allow multiple nodes or multiple VMware ESX servers to read and write to the same LUN or VMFS partition concurrently.

Now that we know about VMFS, let's take a look at the different storage options that are made available.

Direct Attached Storage

Direct-attached storage (DAS) is storage that is, as the name implies, directly attached to a computer or server. DAS is usually the first step taken when working with storage. A good example would be a company with two VMware ESX Servers directly attached to a disk array. This configuration is a good starting point, but it typically doesn't scale very well.

Network Attached Storage

Network-attached storage (NAS) is a type of storage that is shared over the network at a filesystem level. This option is considered an entry-level or low cost option with a moderate performance rating. VMware ESX will connect over the network to a specialized storage device. This device can be in the form of an appliance or a computer that uses Network File System (NFS).

The VMkernel is used to connect to a NAS device via the VMkernel port and supports NFS Version 3 carried over TCP/IP only. From the standpoint of the VMware ESX servers, the NFS volumes are treated the same way VMware ESX would treat iSCSI or Fibre Channel storage. You are able to VMotion guests from one host to the next, create virtual machines, boot virtual machines as well as mount ISO images as CD-ROMs when presented to the virtual machines.

When configuring access to standard Unix/Linux based NFS devices, some configuration changes will need to be defined. The directory /etc/exports will define the systems that are allowed to access the shared directory. And there are a few options in this file that you should be aware of.

1. Name the directory to be shared.
2. Define the subnets that will be allowed access to the share.
3. Allow both "read" and "write" permissions to the volume.
4. no_root_squash—The root user (UID = 0) by default is given the least amount of access to the volume. This option will turn off this behavior, giving the VMkernel the access it needs to connect as UID 0.
5. sync—All file writes MUST be committed to the disk before the client write request is actually completed.

Windows Server 2003 R2 also natively provides NFS sharing when the Windows Services for Unix (SFU) service is installed and configured. Out of the box, Windows Server 2003 R2 has this ability, but it can also be run on Windows Server 2003 (non-R2), and Windows 2000 Server after downloading SFU from Microsoft's Website.

1. After storage has been allocated, the folders are presented similarly as NFS targets.

2. Because there is no common authentication method between VMware ESX and a Microsoft Windows server, the /etc/passwd file must be copied to the Windows server, and mappings must be made to tie an account on the ESX server to a Windows account with appropriate access rights.

Fibre Channel SAN

When using Fibre Channel to connect to the backend storage, VMware ESX requires the use of a Fibre Channel switch. Using more than one allows for redundancy. The Fibre Channel switch will form the "fabric" in the Fibre Channel network by connecting multiple nodes together. Disk arrays in Storage Area Networks (SAN) are one of the main things you will see connected in a Fibre Channel Network along with servers and/or tape drives. Storage Processors aggregate physical hard disks into logical volumes, otherwise called LUNs, each with its own LUN number identifier. World Wide Names (WWNs) are attached by the manufacturer to the Host Bus Adapters (HBA). This is a similar concept as used by MAC addresses within network interface cards (NICs). All Zoning and Pathing is the method the Fibre Channel Switches and SAN Service Processor (SP) use for controlling host access to the LUNs. The SP use soft zoning to control LUN visibility per WWN. The Fibre Channel Switch uses hard zoning, which controls SP visibility on a per switch basis as well as LUN masking. LUN Masking controls LUN visibility on a per host basis.

The VMkernel will address the LUN using the following example syntax:

```
Vmhba(adapter#):target#:LUN#:partition# or Vmhba1:0:0:1
```

So how does a Fibre Channel SAN work anyway? Let's take a look at how the SAN components will interact with each other. This is a very general overview of how the process works.

1. When a host wants to access the disks or storage device on the SAN, the first thing that must happen is that an access request for the storage device must take place. The host sends out a block-based access request to the storage devices.
2. The request is then accepted by the HBA for the host. At the same time, it is first converted from its binary data form to optical form which is what is required for transmission in the fiber optical cable. Then the request is "packaged" based on the rules of the Fibre Channel protocol.
3. The HBA then transmits the request to the SAN.
4. One of the SAN switches receives the request and checks to see which storage device wants to access from the host's perspective; this will appear as a specific disk, but will really be a logical device that will correspond to some physical device on the SAN.

5. The Fibre Channel switch will determine which physical devices have been made available to the host for its targeted logical device.
6. Once the Fibre Channel switch determines the correct physical device, it will pass along the request to that physical device.
7. When a host wants to access the disks or storage device on the SAN, the fist thing that must happen is an access request for the storage device. The host sends out a block-based access request to the storage devices.
8. The request is then accepted by the HBA for the host. At the same time, it is first converted from its binary data form to optical form which is what is required for transmission in the fiber optical cable. Then the request is "packaged" based on the rules of the Fibre Channel protocol.
9. The HBA then transmits the request to the SAN.
10. One of the SAN switches receives the request and checks to see which storage device wants to access from the host's perspective; this will appear as a specific disk but will really be a logical device that will correspond to some physical device on the SAN.
11. The Fibre Channel switch will determine which physical devices have been made available to the host for its targeted logical device.
12. Once the Fibre Channel switch determines the correct physical device it will pass along the request to that physical device.

Internet Small Computer System Interface

Internet Small Computer System Interface or iSCSI is a different approach than that of Fibre Channel SANs. iSCSI is a SCSI transport protocol which enables access to a storage device via standard TCP/IP networking. This process works by mapping SCSI block-oriented storage over TCP/IP. This process is similar to mapping SCSI over Fibre Channel. Initiators like the VMware ESX iSCSI HBA send SCSI commands to "targets" located in the iSCSI storage systems.

iSCSI has some distinct advantages over Fibre Channel, primarily with cost. You can use the existing NICs and Ethernet switches that are already in your environment. This brings down the initial cost needed to get started. When looking to grow the environment, Ethernet switches are less expensive then Fibre Channel switches.

iSCSI has the ability to do long distance data transfers. And iSCSI can use the Internet for data transport. You can have two separate data centers that are geographically apart from each other and still be able to do iSCSI between them. Fibre Channel must use a gateway to tunnel through, or convert to IP.

Performance with iSCSI is increasing at an accelerated pace. As Ethernet speeds continue to increase (10Gig Ethernet is now available), iSCSI speeds increase as well. With the way iSCSI SANs are architected, iSCSI environments continue to

increase in speed the more they are scaled out. iSCSI does this by using parallel connections from the Service Processor to the disks arrays.

iSCSI is simpler and less expensive than Fibre Channel. Now that 10Gig Ethernet is available, the adoption of iSCSI into the enterprise looks very promising.

It is important to really know the limitations and/or maximum configurations that you can use when working with VMware ESX and the storage system on the backend. Let's take a look at the one's that are most important.

1. 256 is the maximum number of LUNs per system that you can use and the maximum during install is 128.
2. There is a 16 port total maximum in the HBAs per system.
3. 4 is the maximum number of virtual HBAs per virtual machine.
4. 15 is the maximum number of targets per virtual machine.
5. 60 is the maximum number of virtual disks per Windows and Linux virtual machine.
6. 256 is the maximum number of VMFS file systems per VMware ESX server.
7. 2TB is the maximum size of a VMFS partition.
8. The maximum file size for a VMFS-3 file is based on the block size of the partition. A 1MB block size will allow up to a 256GB file size and a block size of 8MB will allow 2TB.
9. The maximum number of files per VMFS-3 partition is 30,000.
10. 32 is the maximum number of paths per LUN.
11. 1024 is the maximum number of total paths.
12. 15 is the maximum number of targets per HBA.
13. 1.1GB is the smallest VMFS-3 partition you can create.

So, there you have it, the 13 VMware ESX rules of storage. The setting of the block file size on a partition is the rule you will visit the most. A general best practice is to create LUN sizes between 250GB and 500GB. Proper initial configuration for the long term is essential. An example would be, if you wanted to P2V a server that has 300GB total disk space, and you did not plan appropriately, you would have an issue. Unless you planned ahead when you created the LUN and used a 2MB block size, you would be stuck. Here is the breakdown:

1. 1MB block size = 256GB max file size
2. 2MB block size = 512GB max file size
3. 4MB block size = 1024GB max file size
4. 8MB block size = 2048GB max file size.

Spanning up to 32 physical storage extents (block size = 8MB = 2TB) which equals the maximum volume size of 64TB.

NOTE

Now would be a very good time to share a proverb that has served me well over my career. "Just because you can do something, does not mean you should." Nothing could be truer than this statement. There really is no justification for creating volumes that are 64TB or anything remotely close to that. As a best practice, I start thinking about using Raw Device Mappings (otherwise known as RDMs) when I need anything over 1TB. I actually have 1TB to 2TB in my range, but if the SAN tools are available to snap a LUN and then send it to Fibre tape, that is a much faster way to back things up. This is definitely something to consider when deciding whether to use VMFS or RDM.

System Administrators today do not always have the luxury of doing things the best way they should be done. Money and management ultimately make the decisions, and we are then forced to make due with what we have. In a perfect world, we would design tier-level storage for different applications and virtual machines running in the environment, possibly comprised of RAID 5 LUNS and RAID 0+1 LUNS. Always remember the golden rule—"Spindles equal Speed."

As an example, Microsoft is very specific when it comes to best practices with Exchange and the number of spindles you need on the backend to get the performance that you expect for the scale of the deployment. Different applications are going to have different needs, so depending on the application that you are deploying, the disk configuration can make or break the performance of your deployment.

Summary

So we learned that the number of spindles directly affects the speed of the disks. And we also learned the 13 VMware ESX rules for storage and what we needed to know about VMFS. Additionally, we touched on the different storage device options that have been made available to us. Those choices include DAS, iSCSI and Fibre Channel SAN. We also presented a very general overview on how a Fibre Channel SAN works.

Knowing one of the biggest gotchas is the block size of VMFS partitions and LUNs, and then combining that knowledge with the different storage options made available, you can now make the best possible decisions when architecting the storage piece of your Virtual Infrastructure environment. Proper planning up front is crucial to making sure that you do not have to later overcome hurdles pertaining to storage performance, availability, and cost.