The Global Leader in Scale-out Storage

Isilon IQ and VMware vSphere 4.1

Best Practices for VMware vSphere Using Isilon IQ™ Scale-out Storage

Shai Harmelin, Principal Systems Engineer
Nick Trimbee, Sr. Solutions Architect

An Isilon Best Practices Guide
February 2011

Table of Contents

About this Guide

Intended Audience

This Guide will provide technical information to consider when designing a virtual data center on a foundation of Isilon Scale-out storage. The guide is written for experienced system administrators who are familiar with virtual machine technology and network storage administration.

Assumptions

This Best Practices Guide assumes the reader has:

- Understanding of the iSCSI and NFS Storage Systems
- Working knowledge of Isilon IQ storage and the OneFS® operating system
- Working knowledge of VMware ESX Server and Virtual Center

Additional Resources

Before reading this guide it is recommended to read the following:

- ESX Configuration Guide from VMware
- iSCSI SAN configuration Guide from VMware
- Isilon OneFS 6.0 User Guide from Isilon
- Isilon Best Practices Guide for iSCSI with OneFS from Isilon

Isilon documentations can be found here

http://www.isilon.com/library

VMware documentations can be found here

http://www.vmware.com/support/pubs/vs_pubs.html

1. Isilon Scale-out storage for Server Virtualization

Server virtualization is quickly becoming a standard in major enterprises to simplify overhead and reduce costs of managing large-scale server environments for test, development, and production applications as well as hosted services in the cloud. While virtualization provides a solution for increasing server utilization and reducing operational expenses, the challenges with traditional SAN or NAS storage architectures, storing large numbers of virtual machines, can often negate the potential benefits of server virtualization.

Benefits of Isilon Scale-out storage

The Isilon IQ scale-out storage platform, powered by the OneFS® operating system, was designed from the ground up to reduce storage complexity. In a dynamic virtualized environment, Isilon eliminates the management complexity inherent in traditional SAN and NAS architectures while increasing application availability and protection.

Simplify storage management

Consolidate and replace tens, hundreds, or thousands of volumes with a single namespace, single volume, and single file system to host thousands of virtual machines. With Isilon you can also consolidate virtualized and non-virtualized application storage with standard networking and file sharing protocols.

Quickly adapt to change

In a dynamic environment where virtual machines are added daily, Isilon non-disruptively scales capacity and performance. When a storage node is added to an Isilon cluster the additional capacity, performance and connectivity is immediately available and shared across all datastores.

Increase efficiency and productivity

Isilon can achieve over 80% storage utilization without degrading performance. The globally coherent cache in OneFS combined with the AutoBalance feature, eliminates complex storage management tasks, such as chasing hot-spots (LUN thrashing) and manually migrating virtual machines across volumes, allowing administrators to focus on achieving the greatest gains from a virtualized server environment.

Increase protection and disaster recovery in your virtualized environment

The FlexProtect feature in OneFS, along with a suite of integrated software applications allows non-disruptive protection policy changes, unlimited snapshot and replication schedules at the individual VM level, or across any number of datastores.

OneFS Operating System

The cornerstone of the Isilon fully distributed scale-out architecture is the OneFS Operating System. OneFS combines the three layers of traditional storage architectures — file system, volume manager and RAID — into one unified software layer, creating a single intelligent fully symmetrical file system that spans all nodes within a cluster. In addition to enabling industry-leading scalability of performance and capacity, OneFS combines mission-critical reliability and high availability with state-of-the-art data protection.

Isilon's OneFS operating system stripes both data and its corresponding meta-data across all nodes in a cluster to create a single, shared pool of storage. This approach is a vast improvement over the traditional method of striping data across a limited set of disks (RAID groups). Traditional SAN and NAS architectures use a centralized server to manage a disk array, creating dependencies and multiple points of failure within a storage system. Each node in an Isilon clustered storage system is a peer, so any node can handle I/O requests. OneFS provides each node with knowledge of the entire file system layout giving users access to all content in one unified namespace, with no volumes to manage, no inflexible volume size limits, no downtime required for reconfiguration or expansion of storage and no multiple network drives to manage.

Performance & Scalability*: Isilon eliminates the challenges of scaling workloads and environments.*

Isilon Scale-out storage provides for multi-dimensional scalability of capacity, performance or consolidating applications. Through OneFS the system automatically uses all available storage capacity and compute resources from all nodes (disk, cache, CPU, and network resources), allowing transparent and "on-the-fly" scalability as more capacity and/or performance is needed. The capabilities of Isilon storage are in stark contrast to traditional storage, which rely on single/dual controllers sitting over shelves of disks that are limited in scalability dimensions. A controller will have a pre-defined performance capability (see figure 3 dotted line) which is achieved at some point when additional spindles (shelves of disk) are added to the system. When higher performance or more capacity is needed the primary option is to conduct a "forklift upgrade" of the head.

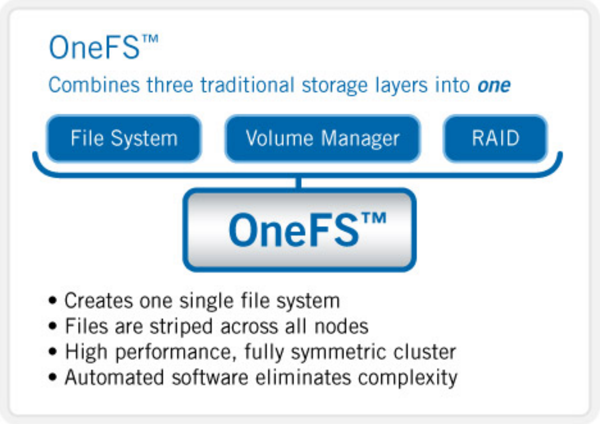

OneFS™
Combines three traditional storage layers into *one*
File System
Volume Manager
RAID
OneFS™
• Creates one single file system
• Files are striped across all nodes
• High performance, fully symmetric cluster
• Automated software eliminates complexity

Figure 1: Consolidating storage layers with OneFS

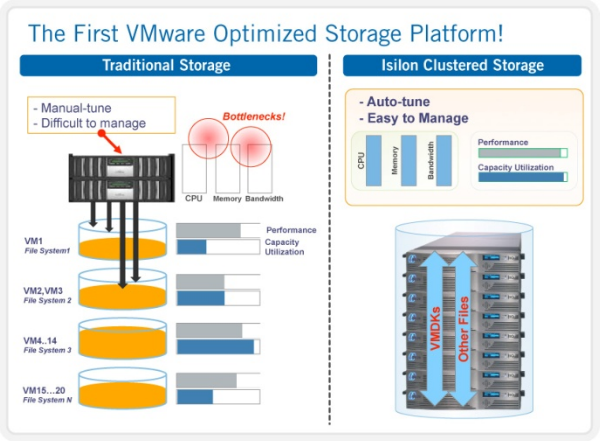
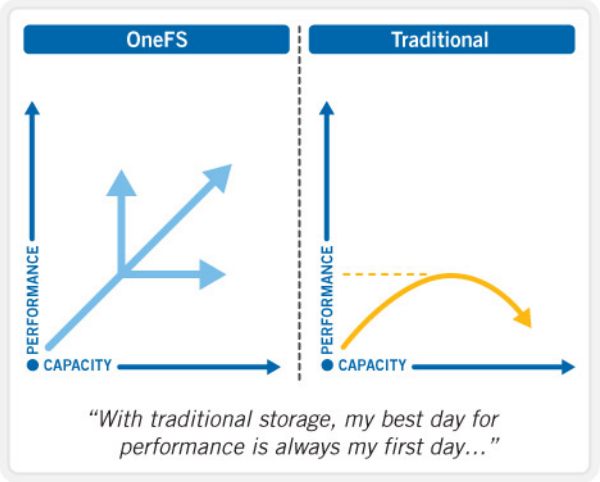


Figure 2: Traditional storage vs. Isilon scale-out storage



"With traditional storage, my best day for performance is always my first day..."

Figure 3: Linear scalability in Isilon scale-out storage

***Availability & Reliability:** OneFS enables Isilon to deliver unprecedented system reliability and data availability.* OneFS FlexProtect™ uniquely stripes data and ECC/parity across all nodes in the cluster (rather than a specific RAID set of disks as with hardware based RAID) which enables Isilon to deliver advanced protection not achievable with traditional storage architectures. An Isilon cluster can survive an unprecedented four (N+4) simultaneous failures of either entire nodes or drives, regardless of the capacity of the drive. In contrast traditional systems can at best sustain only two drive or system failures with traditional RAID protection (e.g. RAID6, RAID DP, N+2, mirroring).

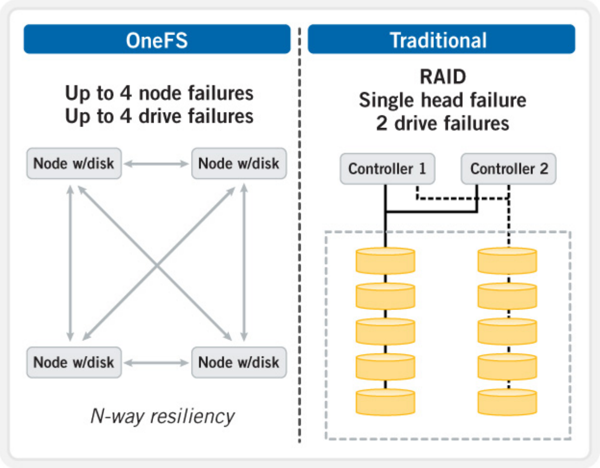


Figure 4: OneFS FlexProtect

FlexProtect™ *Virtual Hot Spare* feature rebuilds failed disks in a fraction of the time compared to traditional RAID. Because the Isilon cluster is a true symmetric system, all nodes in the cluster can participate in rebuilding content from failed disks and harness free storage space across all the spindles in the cluster. As a result Isilon can routinely rebuild data from failed drives in less than an hour with limited, if any performance impact.

The OneFS Operating System constantly monitors the health of all files and disks, processing smart statistics on each drive to anticipate when that drive will fail. When OneFS identifies at risk components, it preemptively and automatchically migrates the data off of the suspect disk to available free space on the cluster, all transpartently to the end-user.

End-to-End Storage Solution for Server Virtualization

Isilon Scale-out storage is purpose-built for file-based and virtualized workflows combining a highly scalable and redundant system that is a simple to manage and simple to scale enterprise storage solution:

- Integration with High Availability, Dynamic Resource Scheduling, VMotion, Storage VMotion, and VADP/VCB
- Isilon recommends NFS-based data stores for ease-of-use and OneFS integration.
- Isilon applications SyncIQ, SnapshotIQ, and combined with FlexProtect in OneFS provide flexible, high performance VM data protection capabilities
- SmartConnect application and AutoBalance in OneFS ensure continual efficient usage of storage resources

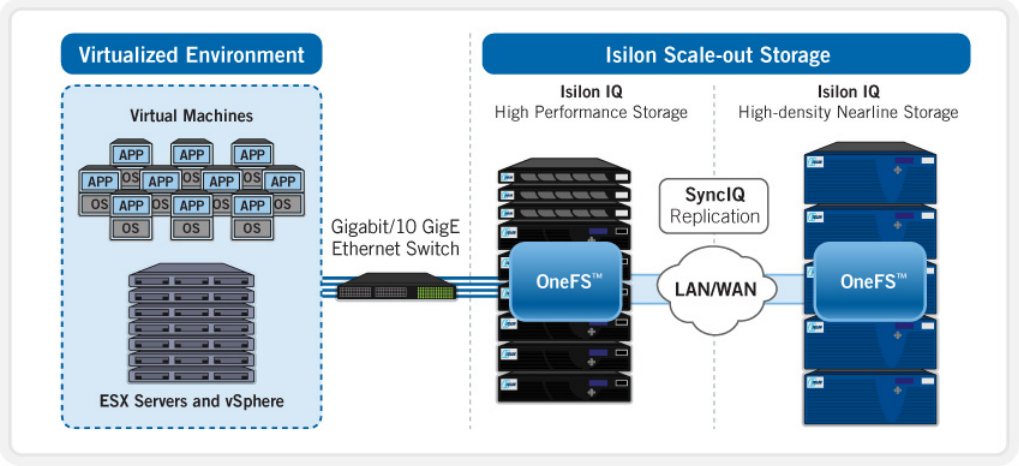


Figure 5: Virtual environment using Isilon scale-out storage

Isilon is VMware Ready

Isilon is VMware Ready certified for both ESX 3.0, vSphere 4 (ESX 4.0 and ESX 4.1). VMware certification is covered across all Isilon product lines, including the X-Series, S-Series, and NL-Series product families. This ensures compatibility with VMware products and that Isilon is ready for deployment in customer environments.

2. VMware Support for Network Storage

A VMware virtual machine uses a virtual hard disk to store the operating system, program files, and other files associated with its applications and user data. Each virtual disk that a virtual machine can access resides in the VMware Virtual Machine File System (VMFS) datastore, NFS-based datastore, or on a raw disk. From the standpoint of the virtual machine, each virtual disk appears as if it were a SCSI drive connected to a SCSI controller. Whether the actual physical disk device is accessed through parallel SCSI, iSCSI, network, or Fibre Channel adapters on the host, it is transparent to the guest operating system and the applications running in it.

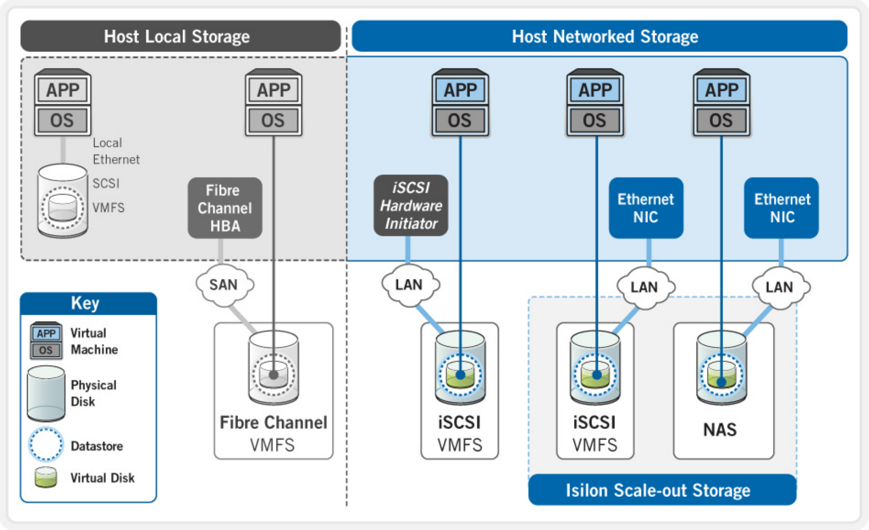


Figure 6: VMware supported storage types

ESX supports the following types of storage:

- **Local Storage:** Stores virtual machine files on internal or external storage disks or arrays attached to the host through a direct connection.
- **Networked Storage:** Stores virtual machine files on external shared storage systems located outside of your host. The host communicates with the networked devices through a high-speed network.

This paper will focus on Ethernet based network storage, namely VMFS datastores over iSCSI and NFS datastores.

Note: Hardware iSCSI adaptors have not been tested as of the release of this paper and are not currently supported for use with Isilon storage.

Network Attached Storage (NAS) for VMware

With ESX 3.0 VMware extended support for NAS using the NFS protocol. Fundamentally, NAS stores large VM datastores on a Network File System (an industry-standard file sharing protocol) export rather than VMFS datastores. The storage system is presented to each ESX host through an NFS mount and ESX hosts access VM data through the NFS protocol. Among the advantages of NAS-based datastores:

- Rapid and simple storage provisioning: Instead of managing LUNs for individual virtual machines, all VMDK files may be stored on a common file export. VMDKs in NFS datastores are thinly provisioned by default and file system storage behind an NFS export is not allocated up front.
- Higher storage utilization rates: VMware disk files (VMDK files) are thin-provisioned by default with NAS datastores. NFS file systems are typically much larger than individual LUNs allowing more virtual machines to share a larger storage pool reducing the overhead of unused storage.
- Simplified backup scenarios: All VM files may be backed up behind a single, central mount point. Individual virtual machines can be protected and backed up simply by backing up the file system directories of those virtual machines.

iSCSI IP Storage Area Network (SAN) for VMware

VMware introduced support for shared storage in ESX Server 2.0. First it provided connectivity to block storage over Fiber Channel and later using the iSCSI protocol over Gigabit Ethernet networks. VMware developed the VMFS shared file system that can share access to iSCSI LUNs across multiple ESX servers to support features such as vMotion and Distributed Resource Scheduling (DRS). iSCSI has seen rapid adoption in virtual environments for the following reasons:

- iSCSI uses the ubiquitous Ethernet network infrastructure to replace more complex Fiber Channel.
- IT administrators can quickly transition to a virtual environment using block storage infrastructure without changes to their applications.
- ESX can leverage iSCSI multi-pathing to provide network redundancy and increase aggregate bandwidth to a single iSCSI LUN and support the aggregate I/O large mix of virtual machines.
- For specific use cases, ESX can provide VMs direct access to iSCSI LUNs including physical-to-virtual-machine host-based clustering, Microsoft Cluster Server (MSCS) or applications that need direct access to a block device. This direct LUN access to a VM is referred to as Raw Device Mapping (RDM) over iSCSI. In this design, ESX acts as a connection proxy between the VM and the storage array.
- A guest OS can use an iSCSI initiator to access a LUN (exclusively). In this mode only higher level networking calls are managed by the virtualization layer while SCSI and iSCSI level commands are handled by the guest OS.
- Ability to clone LUNs either as full clones, shadow clones, or read-only snapshot clones.

While each storage protocol offers its own benefits and limitations the choice of using iSCSI or NFS based datastores often comes down to what the user is already familiar with and the infrastructure available.

Isilon Scale-out Storage for VMware

The following table highlights the capabilities of each protocol supported by Isilon scale-out storage.

	VMFS	VMotion	SVMotion	DRS	HA	Boot VM	Boot ESX	RDM	MSCS	Multi-Path	VCB	SRM
iSCSI *	Yes	Yes	Yes	Yes	Yes	Yes	No*	Yes	Yes	MPIO**	Yes	No
NFS	NA	Yes	Yes	Yes	Yes	Yes	No	No	No	LAG	Yes	No

Table 1: VMware supported features per storage protocol

Capability	iSCSI *	NFS
File System	VMFS or RDM	OneFS
Max # ESX Datastores	256	64
Max ESX Datastore Size	64TB	10PB
Max LUN/File System Size	32TB ***	10PB
Recommended # of VMs per LUN/File System	20/LUN	Thousands
Network Bandwidth	1G/10GbE	1G/10GbE

Table 2: VMware storage capabilities

*Isilon currently supports ESX Software iSCSI initiators only.
** iSCSI MPIO was introduced in ESX 4.0. Before that LAG was used for multi-pathing.
***ESX can only see up to 2TB size LUNs.

General Networking Guidelines for Ethernet/IP based ESX Datastores

Since this guide focuses on Ethernet/IP-based network storage it is important the network infrastructure is designed and implemented to support production scale, mission-critical, virtual environments. To ensure quality of service and ability to survive network failures, the same design thinking applied to Fiber Channel infrastructure where network traffic-to-storage is isolated and redundant, must be applied to Ethernet/IP storage.

Some important considerations common to both iSCSI and NFS datastore storage networking include:

- Separate storage and VM network traffic to different physical network ports. This separation will avoid having a temporarily busy LAN overrun Ethernet storage traffic. It is also recommended to have a separate vSwitch for VM network, vMotion, and storage networking.
- Consider using switches that support “cross-stack Etherchannel” or “virtual port channel” trunking where interfaces on different physical switches are combined into an 802.3ad LAG for network and switch redundancy.
- Because each Ethernet switch varies in its internal architecture choose a production quality switch for mission-critical, network intensive purposes like VMware datastores (on iSCSI or NFS) that has the required amount of port buffers, and other internal resources.

- 10GbE infrastructure is a great way to consolidate network ports but standard Gigabit infrastructure can meet most ESX network storage needs. For Gigabit infrastructure use Cat6 cables rather than Cat5/5e.
- Consider using Flow-Control. ESX hosts and the Isilon cluster should be set to **send on** and **receive off**. The switch ports connected to the ESX host and Isilon cluster nodes should be set to **send off** and **receive on**.
- For ESX hosts that have fewer NICs (such as blade servers), VLANs can group common IP traffic onto separate VLANs for optimal performance and improved security. VMware recommends to separate **service console** access and virtual machine network from the **VMkernel** network for **IP storage** and **vMotion** traffic.
- Isilon supports restricting NFS connections to specific nodes and network interfaces that are connected to a specific VLAN. If you are using multiple VLANs over the same interface make sure sufficient throughput can be provided for all traffic.

ESX provides network storage high-availability using multiple network interfaces on the local ESX host. The actual configuration of those interfaces varies based on the storage protocol used and the capabilities of the switches connecting the ESX hosts and the Isilon storage. Those configuration options are discussed in more detail within the iSCSI and NFS multi-pathing network configuration sections to follow.

Review of OneFS Networking Configuration

SmartConnect™ is the OneFS technology used for configuring and managing the Isilon network interfaces. With the introduction of OneFS 6.0, major improvements were made to SmartConnect, now in version 2.0. SmartConnect supports complex and variable network topologies and hierarchical and overlapping management objects that allow for extremely flexible configurations, simply defined and managed. The following terms are important for understanding the operation of SmartConnect:

1. **Subnet** – Specifies a network subnet, netmask, gateway and other parameters related to layer-3 networking. (VLAN tagging is configured here). A subnet contains one or more pool objects, which assign a range of IP addresses to a selected set of network interfaces on the Isilon cluster nodes.
2. **Pool** – Also referred to as an IP Address Pool, the Pool contains one or more network interfaces (e.g. Node1:Ext-1) and a set of IP addresses to be assigned to them. Isilon SmartConnect Advanced™ settings, such as the zone name and allocating IPs (statically or dynamically), are also configured at the Pool level.
3. **Provisioning Rule** – Based on the node type and interface, provisioning rule specifies subnet and IP pool assignment actions when a node is added to the cluster. For example, a rule could state when an Isilon IQ storage node is added, External-1 and External-2 are assigned to two different pools, which in turn belong to two separate subnets.

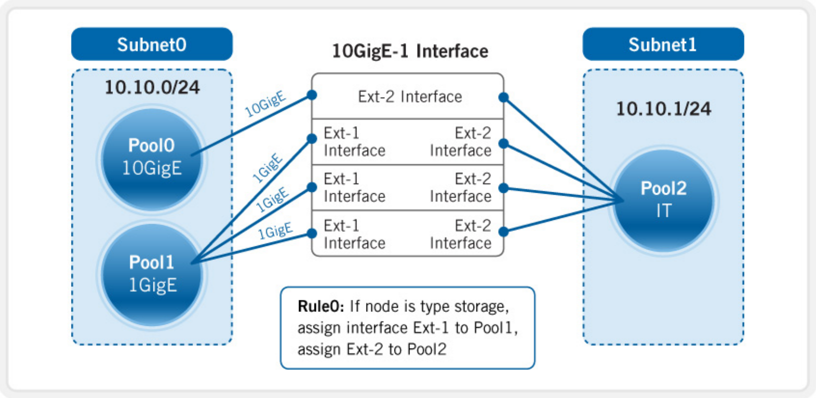


Figure 7: SmartConnect IP pools, subnets and rules

3. Isilon IQ Cluster Configuration for NFS Datastores

The Isilon storage platform is a “Scale-out” approach to NAS storage and offers significant reductions in management overhead by managing the cluster through a single, common management point. In an Isilon cluster, terabytes to petabytes of storage can be administered in a single file system, as compared to other storage architectures that are managed in many small volumes of storage. With a single file system and a single NFS export, Isilon Scale-out storage can accommodate any number of ESX datastores and virtual machines. As follows in this section most of the configuration work focuses on building a robust and redundant network infrastructure while very little configuration is required from the OneFS file system. This highlights how using NFS datastore can greatly simplify VMware storage management.

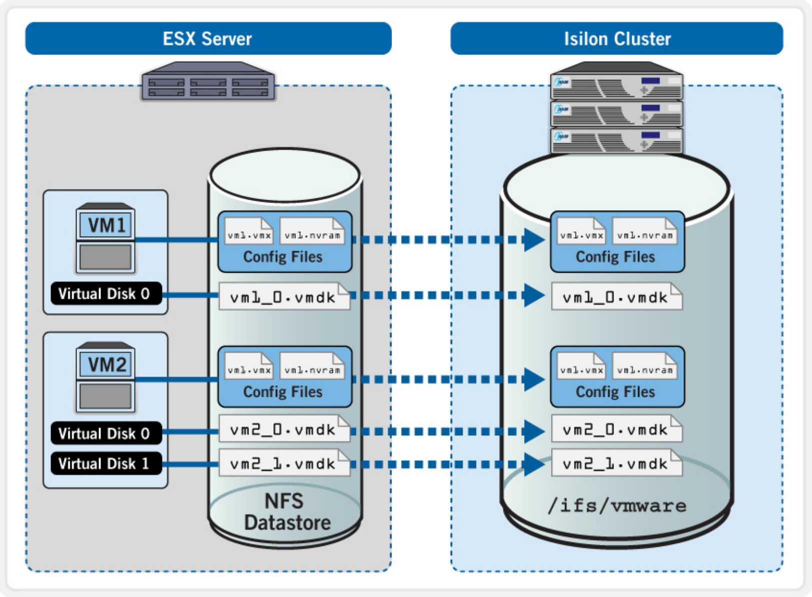


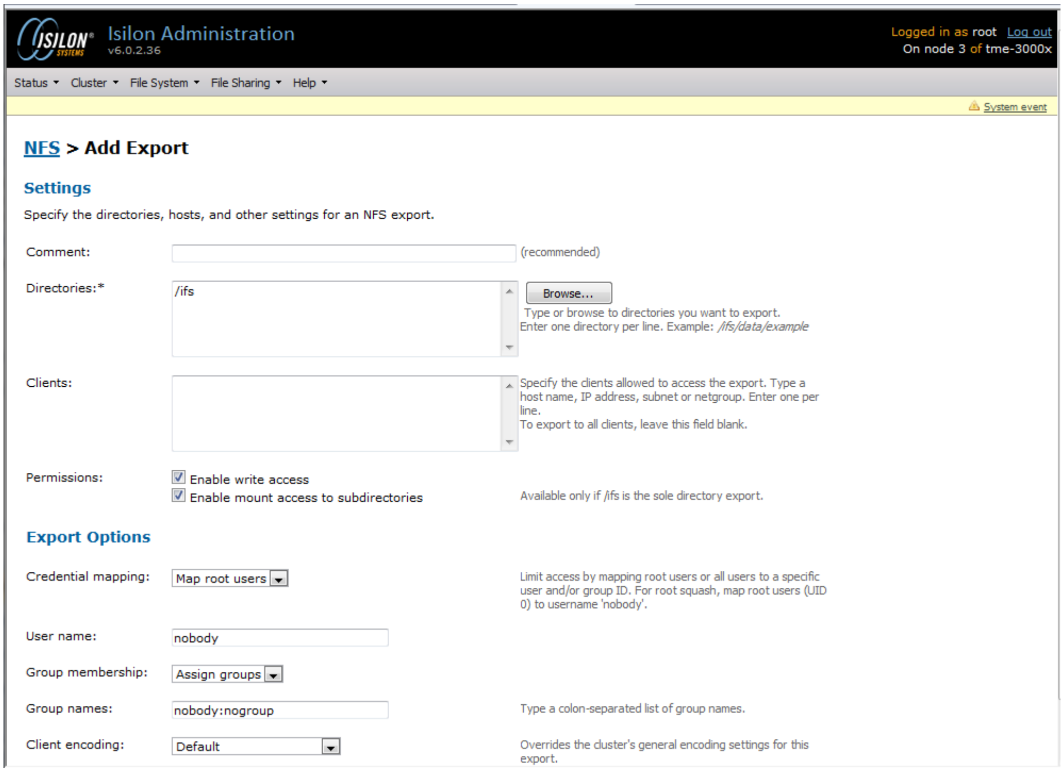
Figure 8: ESX datastore on Isilon IQ cluster

Isilon OneFS File System Configuration

OneFS NFS Exports

Once an Isilon cluster is built (with a minimum of three nodes) all capacity is fully and thinly provisioned and available for NFS access from ESX hosts. ESX hosts can access the shared storage from any assigned IP address on the cluster and the default root /ifs directory, or any subdirectory, can be mounted. A single NFS export can support any number of datastores from the virtual data center since each datastore can access the OneFS file system across different network paths using a different IP address. To create or edit an NFS export:

1. From the WebUI select **File System→File Sharing Services→Configure NFS**
2. Select **Add Export** or **Edit** an existing export.
3. Select the path to export.
4. Select **Enable write access** and **Enable mount access to sub-directories** (only available for /ifs export).
5. Map root **user** to user **nobody** (also known as root squash)
6. Click Submit to exit and save.



NFS > Add Export

Settings

Specify the directories, hosts, and other settings for an NFS export.

Comment: (recommended)

Directories:* /ifs — Type or browse to directories you want to export. Enter one directory per line. Example: /ifs/data/example

Clients: — Specify the clients allowed to access the export. Type a host name, IP address, subnet or netgroup. Enter one per line. To export to all clients, leave this field blank.

Permissions: ☑ Enable write access
☑ Enable mount access to subdirectories — Available only if /ifs is the sole directory export.

Export Options

Credential mapping: Map root users — Limit access by mapping root users or all users to a specific user and/or group ID. For root squash, map root users (UID 0) to username 'nobody'.

User name: nobody

Group membership: Assign groups

Group names: nobody:nogroup — Type a colon-separated list of group names.

Client encoding: Default — Overrides the cluster's general encoding settings for this export.

Figure 9: Configuring an NFS export

When an ESX NFS datastore is created, the directory where the datastore will mount OneFS must already exist. By default, root access to the cluster over NFS is prevented by mapping the ‘root’ user to the user ‘nobody. If the exported directory is created by root, and the ownership isn’t changed, the ESX Server(s) can’t write to the directory. To avoid write access restrictions, simply create a datastore to the /ifs root directory (or /ifs/data) and create a new directory for the datastore by browsing through the datastore explorer. Next, you need to redefine the datastore on the newly created directory. If the directory was created directly on the Isilon cluster using the ‘root’ user, write access can be enabled by using ‘chown’ to change the VMs directory owner to ‘nobody’:

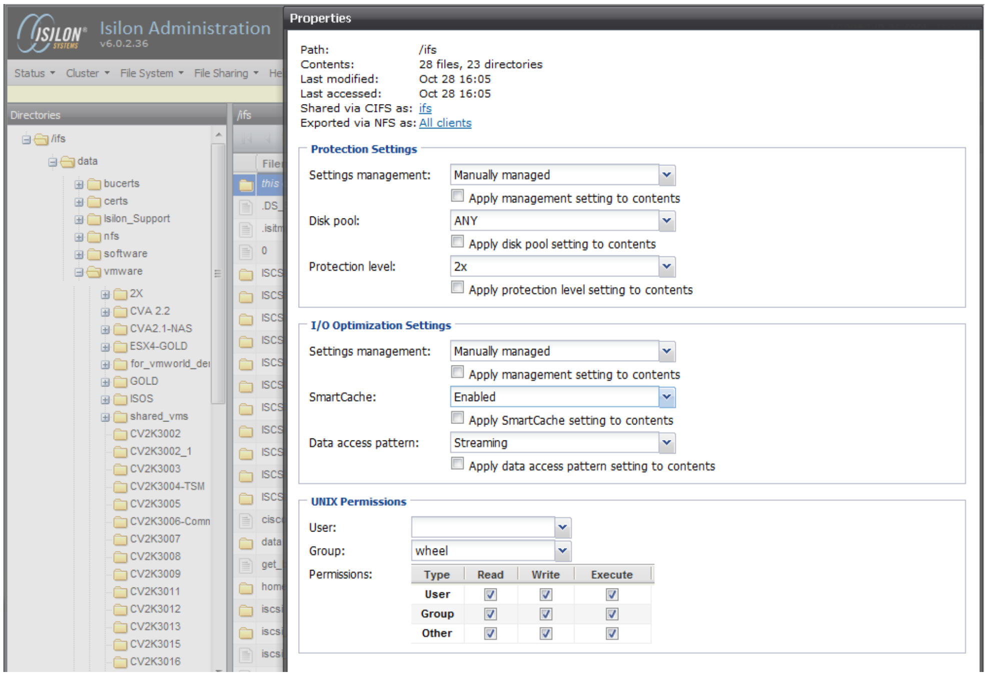
```
‘chown nobody:wheel <directory>’
```

Optimizing OneFS for NFS Datastores with I/O Intensive Virtual Machines

1. By default, OneFS protects file data by calculating and writing parity blocks for each protection group. Virtual machines that exhibit performance issues with small random I/O operations (less than 32K) may benefit from a mirroring data layout (2X) on VM directories and their sub-directories. This setting increases protection overhead and decreases write overhead. The setting can also be applied on a per VM directory basis while other VM directories continue to use the parity protection layout which is more space efficient.
2. OneFS ‘**streaming’** mode on virtual machines with high read and write I/O requirements should be enabled. This setting applies to random and sequential I/O workloads and can be set in conjunction with mirroring layout.
3. Disable OneFS **read caching (prefetch)** if many virtual machines require high ratio of small random read operations. Disabling **read prefetch** instructs OneFS to avoid prefetching adjacent file blocks and will eliminate prefetch latency overhead.

To change a VM directory write data protection and access mode:

1. From the WebUI select **File System→File System Settings->Flex Protect Policy** and set to **Advanced**
2. From the WebUI **select File System→File System Explorer** and use navigate to the VM desired VM directory
3. Select **2X Protection Level** and **apply protection to contents**
4. Select **Optimize writing for Streaming** and **Apply optimization to contents**



Properties

Path: /ifs
Contents: 28 files, 23 directories
Last modified: Oct 28 16:05
Last accessed: Oct 28 16:05
Shared via CIFS as: ifs
Exported via NFS as: All clients

Protection Settings

Settings management: Manually managed
□ Apply management setting to contents
Disk pool: ANY
□ Apply disk pool setting to contents
Protection level: 2x
□ Apply protection level setting to contents

I/O Optimization Settings

Settings management: Manually managed
□ Apply management setting to contents
SmartCache: Enabled
□ Apply SmartCache setting to contents
Data access pattern: Streaming
□ Apply data access pattern setting to contents

UNIX Permissions

User:
Group: wheel
Permissions:

Type	Read	Write	Execute
User	☑	☑	☑
Group	☑	☑	☑
Other	☑	☑	☑

Figure 10: VM directory write access settings for increasing small random I/O

To disable cluster read prefetch setting:

1. Log on to any of the nodes in the cluster over SSH connection
2. At the login prompt issue the following command: `sysctl efs.bam.enable_prefetch=0`
3. To make this setting persistent across cluster reboots, add the following line to the file /etc/mcp/override/sysctl.conf: `efs.bam.enable_prefetch=0`

Isilon Network Configuration Best Practices for ESX NFS Datastores

Since iSCSI and NFS protocols both work over Ethernet/IP infrastructure they share some common design guidelines and best practices. Those best practices were described in Section #2 of the paper "**General Networking Guidelines for Ethernet/IP based ESX Datastores"**. However, NFS and iSCSI have different mechanisms when using multiple network paths to achieve network redundancy and increase aggregate network throughput.

VMware Network Redundancy Options for NFS

For NFS datastores, redundancy is achieved using the link aggregation capabilities provided by the switch and end points (ESX server and Isilon cluster nodes).

- If the switches support “cross-stack Etherchannel” (or “virtual port channel” trunking) in which two ports from two different switches can be aggregated to form a single network trunk (within a single subnet), ESX servers can be configured with one VMKernel port with multiple active network interfaces in a team. To use both interfaces concurrently, create multiple NFS datastore IPs on the same subnet to the target Isilon cluster and set ESX to use source-destination IP load balancing.
- If the switches only support Etherchannel (or “virtual port channel” trunking) within a single switch and two switches are used for redundancy (with two separate subnets), then two or more separate VMKernel ports are required in separate vSwitches. Each VMkernel port has one active, and one (or more) passive network interfaces. Each VMkernel port is on a separate subnet, one for each of the redundant switches.

In both scenarios each of the interfaces in a network aggregate (on the ESX host and the Isilon node) are physically connected to each of the two switches. The Isilon cluster can be configured to use dynamic IPs across multiple nodes for each subnet, or link aggregation, within a single node for each subnet.

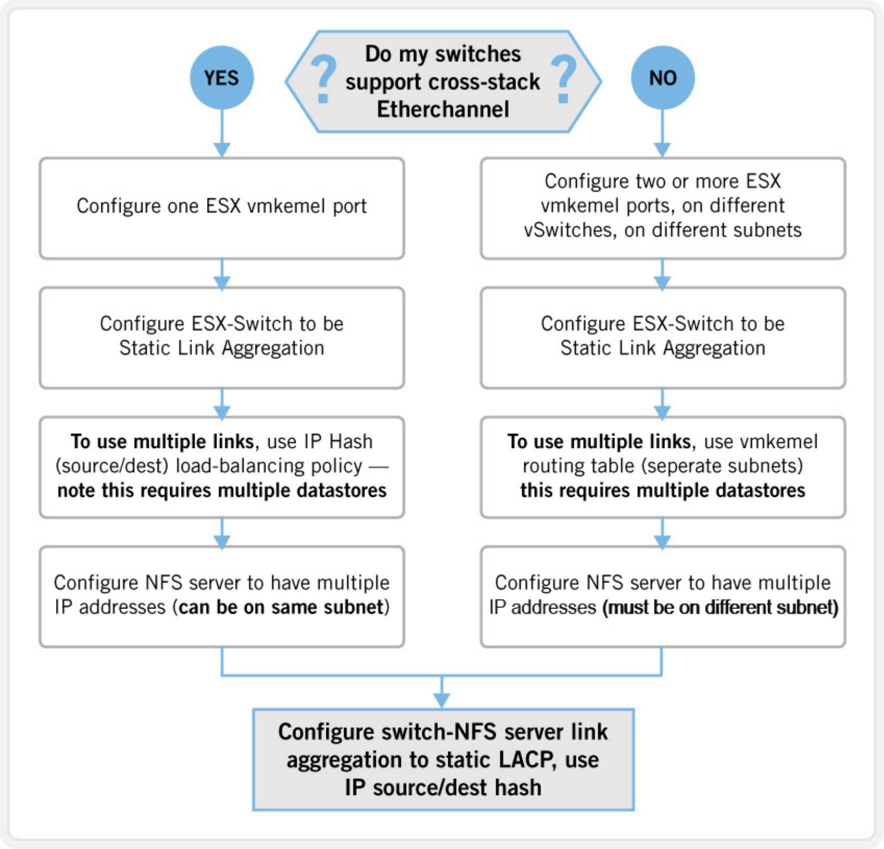

YES
Do my switches support cross-stack Etherchannel
NO
Configure one ESX vmkemel port
Configure two or more ESX vmkemel ports, on different vSwitches, on different subnets
Configure ESX-Switch to be Static Link Aggregation
Configure ESX-Switch to be Static Link Aggregation
To use multiple links, use IP Hash (source/dest) load-balancing policy — note this requires multiple datastores
To use multiple links, use vmkemel routing table (seperate subnets) this requires multiple datastores
Configure NFS server to have multiple IP addresses (can be on same subnet)
Configure NFS server to have multiple IP addresses (must be on different subnet)
Configure switch-NFS server link aggregation to static LACP, use IP source/dest hash

Figure 11: Network redundancy for NFS datastores

Isilon recommends the following to maximize performance, flexibility and availability for NFS datastores:

1. Consider assigning at least one dynamic IP for each member node interface in a dynamic IP pool. Each unique IP can be mounted by a separate ESX NFS datastore, and the more NFS datastores are created, the more TCP connections an ESX host can leverage to balance VM I/O.
2. OneFS Dynamic IPs and Cisco cross-stack EtherChannel link aggregation can be combined to provide both node/interface redundancy and switch redundancy.
3. You can choose to assign a single IP to an aggregated pair of network interfaces (using LAG 802.3d link aggregation) or, assign both interfaces as independent members of a SmartConnect dynamic IP pool. Both options employ multiple network interfaces on each node that are used for redundancy and/or to aggregate I/O. Note: High-availability using dynamic IPs is described later in this chapter. Nodes that have four interfaces can be first link aggregated by pairs (and each linked pair can be assigned its own static IP) or, be members of a dynamic IP pool.

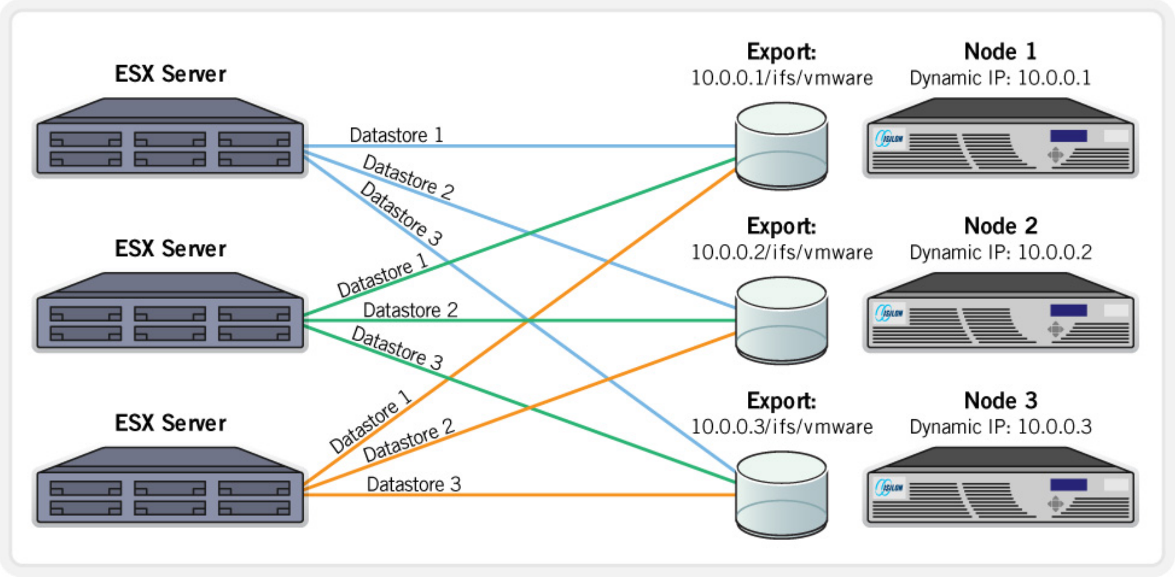


Figure 12: Multiple datastores on a single OneFS NFS export

As Figure 12 illustrates, the best design is a mesh connectivity topology, in which every ESX server is connected to every IP address on a cluster-dynamic IP pool (limited by the maximum amount of NFS datastores as described on page 15). Connecting “everything to everything” enables the following:

- By definition all ESX hosts are connected to all datastores, enabling vMotion between all ESX servers to be performed, knowing all servers can see the same datastore and that the migration will be successful.
- Virtual machines can be created on any datastore to balance the I/O load between ESX servers and the cluster; virtual machines can be easily moved between datastores to eliminate hot spots without moving VM data.

Basic Network Configuration for NFS Datastores

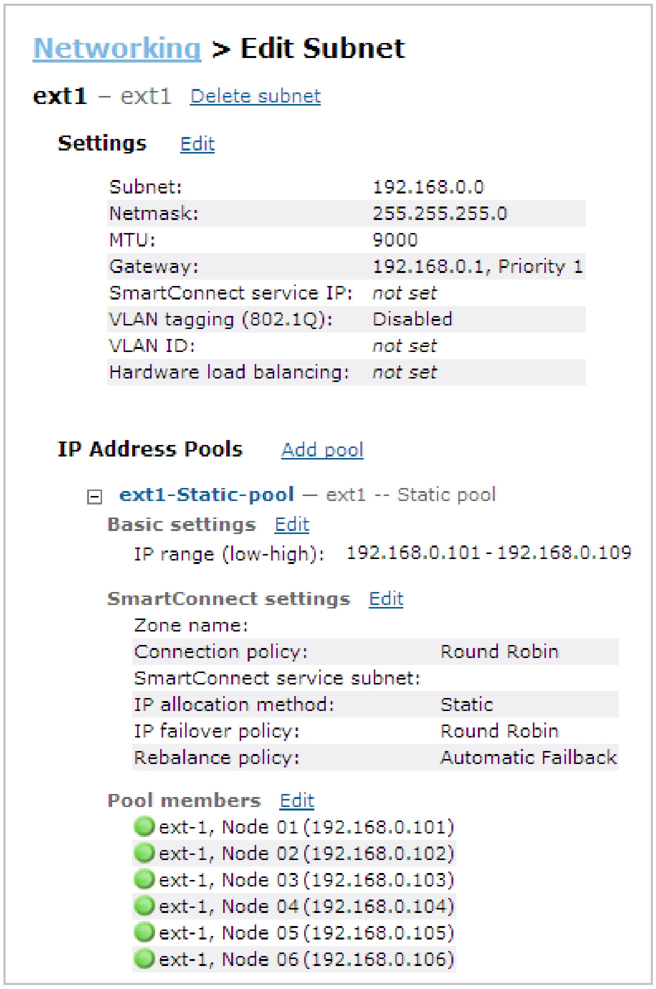
When initially configuring a cluster, the first external network interface (typically External-1) is setup as part of the configuration wizard process. In order for this process to complete successfully, the following information is required: Netmask, IP address range, Default gateway, Domain name server list (optional), DNS search list (optional)

When this information is provided, the following occurs:

- A default external subnet is created, named *subnet0*, with the specified netmask.
- A default IP address pool is created, named *pool0*, with the specified IP address range
- The first node in the cluster is assigned an IP from the address pool and set with the specified default gateway and DNS settings.
- A default network provisioning rule is created, named *rule0*, which automatically assigns the first external interface for all newly added nodes to *pool0*.

After initial cluster configuration is committed, additional subnets and pools can be added or edited.

Figure 13 shows the Edit Subnet page from the WebUI.



Networking > Edit Subnet

ext1 – ext1 Delete subnet

Settings Edit

Subnet:	192.168.0.0
Netmask:	255.255.255.0
MTU:	9000
Gateway:	192.168.0.1, Priority 1
SmartConnect service IP:	*not set*
VLAN tagging (802.1Q):	Disabled
VLAN ID:	*not set*
Hardware load balancing:	*not set*

IP Address Pools Add pool

⊟ **ext1-Static-pool** — ext1 -- Static pool

Basic settings Edit

IP range (low-high): 192.168.0.101 - 192.168.0.109

SmartConnect settings Edit

Zone name:	
Connection policy:	Round Robin
SmartConnect service subnet:	
IP allocation method:	Static
IP failover policy:	Round Robin
Rebalance policy:	Automatic Failback

Pool members Edit

● ext-1, Node 01 (192.168.0.101)
● ext-1, Node 02 (192.168.0.102)
● ext-1, Node 03 (192.168.0.103)
● ext-1, Node 04 (192.168.0.104)
● ext-1, Node 05 (192.168.0.105)
● ext-1, Node 06 (192.168.0.106)

Figure 13: WebUI subnet configuration

VLAN Tagging

Virtual LANs (VLANs) are used to logically group together network endpoints, and to partition network traffic, e.g. for security. VLANs are tagged with a unique identifier to segregate traffic. SmartConnect 2.0 supports VLAN tagging for use in external networks using VLANs. In SmartConnect, VLANs are configured at the Subnet level.

High-Availability Using Link Aggregation

Isilon OneFS supports the use of redundant NICs to provide layer-2 failover. OneFS link aggregation supports the IEEE 802.3ad static LAG protocol, and works with switches and clients that support this protocol.

Note: Isilon does not currently support dynamic LACP.

Link Aggregation Switch Support

Isilon network link aggregation requires 802.3ad LAG support and proper configuration on the switch. Note: Cisco switches support this using the EtherChannel feature. It is strongly recommended to configure cross-stack EtherChannel for protection against switch failures as well as NIC failures.

Link Aggregation Cluster Configuration

Link aggregation, can be configured on the Isilon IQ cluster for a new subnet (or an existing one) by creating an IP pool with the aggregated interface on each node as the pool's members:

1. On the **Edit Subnet** page, at the top of the **IP Address Pools** section, click the **Add pool** link.
2. In the **Create Pool** wizard, enter a name for the pool, and optional description, and a range of IP addresses to use for this pool. Click Next
3. If SmartConnect is used, options for the pool can be set on the next page of the wizard. Once these options have been selected, click Next.
4. On the next page, the interfaces to be members of this pool are selected. To use link aggregation, select the **'ext-agg'** interface for each node to be in the pool. The interface type is also listed as AGGREGATION.
5. Click Submit to complete the wizard.

Note: Link aggregation provides protection against NIC failures but does not increase performance. If configured properly on the ESX host, both NICs in the aggregate can be used for VMware datastores I/O, but the two channels are not 'bonded' into a single 2 Gigabit link. Each NIC is serving a separate TCP connection.

High Availability Using SmartConnect Advanced Dynamic IP for NFS Failover

How NFS Failover Works

SmartConnect Advanced implements NFS failover by assigning one or more dynamic IP addresses to each node's member interface in the cluster from a configured range of addresses. If a single interface or an entire node experiences a failure, SmartConnect reassigns the dynamic IP addresses to the remaining pool member interfaces. All datastore I/O is immediately re-routed to the newly assigned member interface and VM virtual disk access continues without interruption.

It is recommended to set VMware NFS datastores with dynamic IP addresses assigned to the Isilon storage cluster and not the static IP addresses because static IPs on member nodes interfaces do not get reassigned in the event of an interface or node failure.

When either an interface or the entire node is brought back online, the dynamic IPs in the pool are redistributed across the new set of pool members. This failback mechanism can occur automatically or manually and happens without datastore traffic interruption.

Figure 14 illustrates NFS datastore redundancy using dynamic IPs for NFS failover

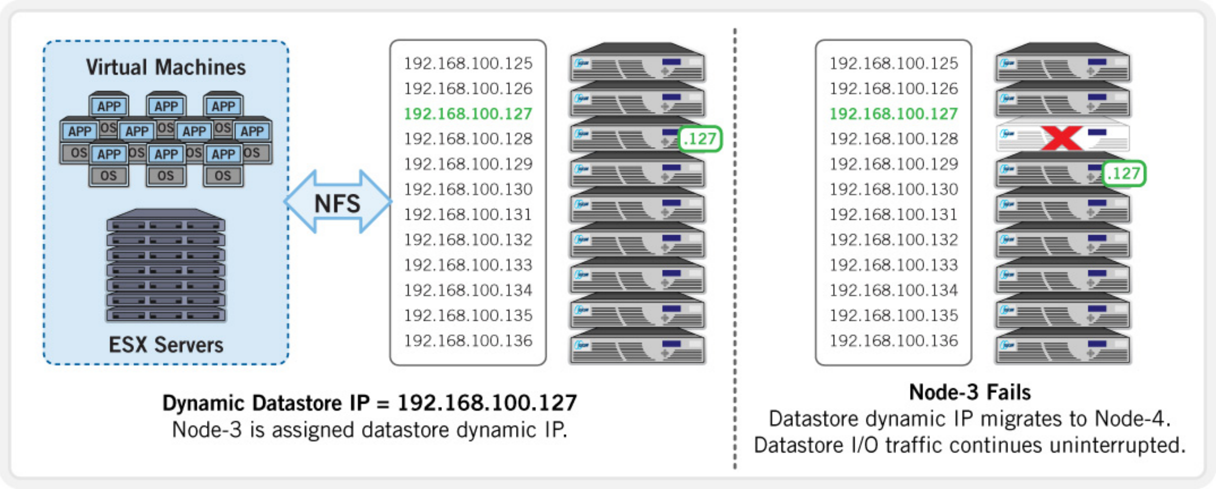


Figure 14: Example of a NFS failover scenario

Limitations of using VMware vSphere with Isilon SmartConnect

Due to the way VMware vCenter manages datastore location paths, it does not support a DNS infrastructure in which a hostname is bound to multiple IP addresses (to allow multiple IP paths to the same datastore). This means datastores must be created using the IP addresses rather than a DNS host name and SmartConnect connection load-balancing does not work.

However, this limitation does not exclude the use of dynamic IP addresses to implement NFS failover for NFS datastores hosted on the Isilon cluster.

4. ESX Configuration for NFS Datastores

This section details the steps necessary to configure ESX Server for use with Isilon storage. Follow these steps to configure a network between the ESX server machine and the Isilon cluster.

The NFS client built into ESX allows you to access the NFS server to use the remote file system to store VM images, ISO images, and templates. To accomplish this, you use the vSphere Client to configure NFS volumes as datastores. Configured NFS datastores appear in the vSphere Client, and you can use them to store virtual disk files in the same way you use VMFS-based datastores.

Note: ESX supports only NFS Version 3 over TCP.

Best Practices for ESX Network Configuration

Isilon recommends the following ESX network configuration for NFS datastores:

1. Create the virtual switch using at least one dedicated network interface card (NIC) for network storage traffic. This will ensure good storage I/O performance, as well as isolate any problems caused by other network traffic.
2. For network redundancy on the ESX host, Isilon recommends teaming multiple network interfaces to the vSwitch of the VMkernel port used for NFS storage. Multiple interfaces in the same vSwitch can also be used to increase the aggregate throughput through the VMkernel NFS storage stack (as long as multiple datastores are assigned different IP addresses). The VMkernel will use multiple TCP/IP connections across multiple network interfaces to provide parallel virtual machine I/O access.

ESX Network Configuration

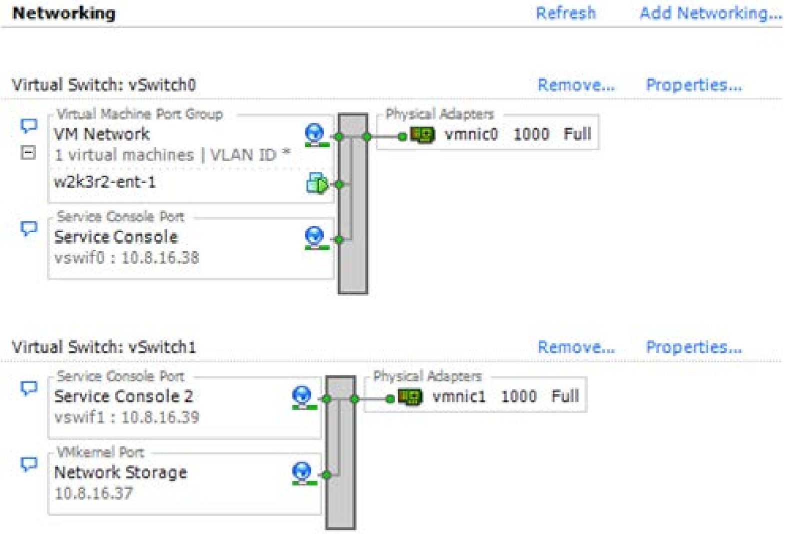
Because NFS requires network connectivity to access data stored on remote servers, before configuring NFS, you must first configure VMkernel networking.

Creating a Virtual Switch

The first step is to create a virtual switch for all network traffic between the ESX server machine and the Isilon cluster.

1. In the vSphere Client, select the ESX server machine in the left-side tree view, then select the **Configuration** tab in the right-side pane.
2. Under **Hardware**, select **Networking**, then select **Add Networking**.
3. In the **Add Network Wizard**, in the **Connection Types** section, select **VMkernel**, and click **Next**.
4. On the **Network Access** screen, select **Create a virtual switch**, or select an existing virtual switch. Click **Next**. To ensure good performance, and isolate any problems from other traffic, Isilon recommends using at least one dedicated network interface card (NIC) for network storage traffic.
5. On the **Connection Settings** screen, enter a network label and optional **VLAN ID**. It's often helpful to give the virtual switch a meaningful label, such as **"NFS Storage".**
6. In the **IP Settings** section, enter an IP address and subnet mask for the **VMkernel port**.
7. If necessary, click the **Edit** button to change the default gateway. Click **Next** to go to the Summary screen.
8. On the **Summary** screen, review the settings, and if correct, click **Finish**.

Figure 15 provides an example configuration with virtual machine and VMkernel networks using separate physical NICs.



Networking

Refresh Add Networking...

Virtual Switch: vSwitch0

Remove... Properties...

Virtual Machine Port Group
VM Network
1 virtual machines | VLAN ID *
w2k3r2-ent-1

Service Console Port
Service Console
vswif0 : 10.8.16.38

Physical Adapters
vmnic0 1000 Full

Virtual Switch: vSwitch1

Remove... Properties...

Service Console Port
Service Console 2
vswif1 : 10.8.16.39

VMkernel Port
Network Storage
10.8.16.37

Physical Adapters
vmnic1 1000 Full

Figure 15: Example ESX network configuration

Configuring a Service Console (ESX 3.5 Only)

Important: ESX 3.5 requires you to configure a service console on the virtual switch you just created. Without a service console, it is possible for the ESX server machine to lose connectivity to storage located on the virtual switch. This step is NOT necessary for vSphere and ESX 4.1

1. In the VI Client, on the **Configuration** tab for the ESX server machine, select **Properties** next to the virtual switch that you just created.
2. In the **Properties** dialog, on the **Ports** tab, click **Add**.
3. In the **Add Network Wizard**, in the **Connection Types** section, select **Service Console**, and click **Next**.
4. On the **Connection Settings** screen, enter a network label and optional **VLAN ID**.
5. The console can be given a static IP address or obtain one via DHCP, then click **Next**.
6. On the **Summary** screen, review the settings, and if correct, click **Finish**.

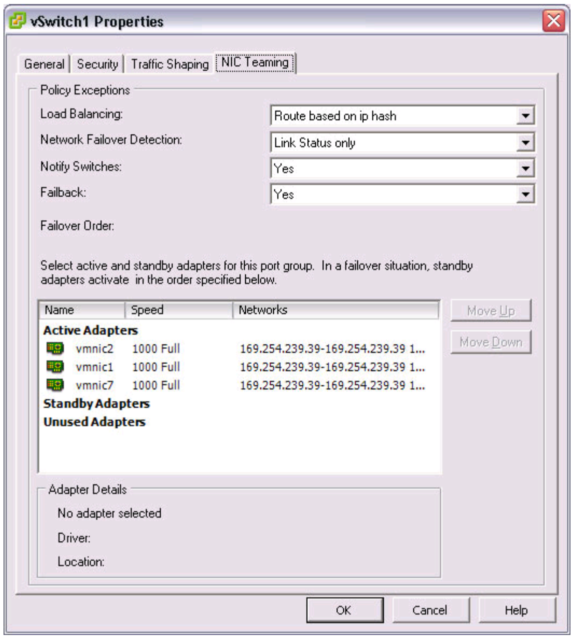
Configuring Link Aggregation

Link aggregation, also known as NIC failover, or NIC teaming, is one approach to ensure higher network availability between the ESX server and Isilon cluster. NIC teaming is a layer-2 IEEE standard known as 802.3ad. Perform the following steps to configure NIC teaming on an ESX server.

Note: NIC teaming requires that both NICs involved in the team are on the same subnet.

1. If two NICs are not configured in the virtual switch, add a second NIC by selecting **Properties** on the **Configuration** tab for the ESX Server.
2. On the **Properties** dialog, select the **Network Adapters** tab, and click **Add**. Follow the instructions in the **Add Adapter** wizard.
3. Once the second NIC is added to the virtual switch, **teaming** is enabled using a default configuration. To change **NIC teaming** options, select **Properties**... for the virtual switch.
   - On the **Properties** dialog, select the **port group** (“NFS Storage” in this example), and click **Edit**.
   - On the **port group properties** dialog, select the **NIC Teaming** tab to change the configuration.
4. Isilon recommends setting the following NIC teaming options:
   - Network Failover Detection: Set to Link Status.
   - Notify Switches: Yes (typically the default).

   In a failover event, a notification is sent out over the network to update the lookup tables on physical switches, which will reduce the latency associated with failover.
5. Isilon recommends setting the following load balancing options: **Route based on IP hash**
6. Click **OK** to exit the port group properties dialog, then Close to exit the virtual switch properties dialog.


vSwitch1 Properties

Policy Exceptions

Load Balancing: Route based on ip hash

Network Failover Detection: Link Status only

Notify Switches: Yes

Failback: Yes

Failover Order:

Select active and standby adapters for this port group. In a failover situation, standby adapters activate in the order specified below.

Name	Speed	Networks
Active Adapters		
vmnic2	1000 Full	169.254.239.39-169.254.239.39 1...
vmnic1	1000 Full	169.254.239.39-169.254.239.39 1...
vmnic7	1000 Full	169.254.239.39-169.254.239.39 1...
Standby Adapters		
Unused Adapters		

Adapter Details

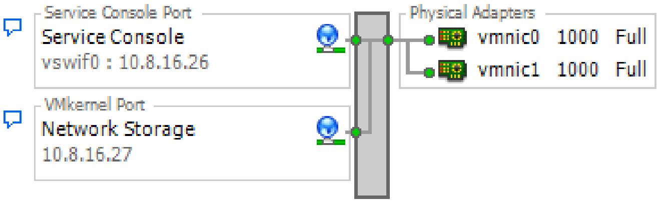
No adapter selected

Driver:

Location:

Figure 16: NIC teaming configured on ESX host

After adding the second NIC, the virtual switch diagram will look like Figure 17.


Service Console Port

Service Console

vswif0 : 10.8.16.26

VMkernel Port

Network Storage

10.8.16.27

Physical Adapters

vmnic0	1000	Full
vmnic1	1000	Full

Figure 17: Multiple NICs in a Virtual Switch

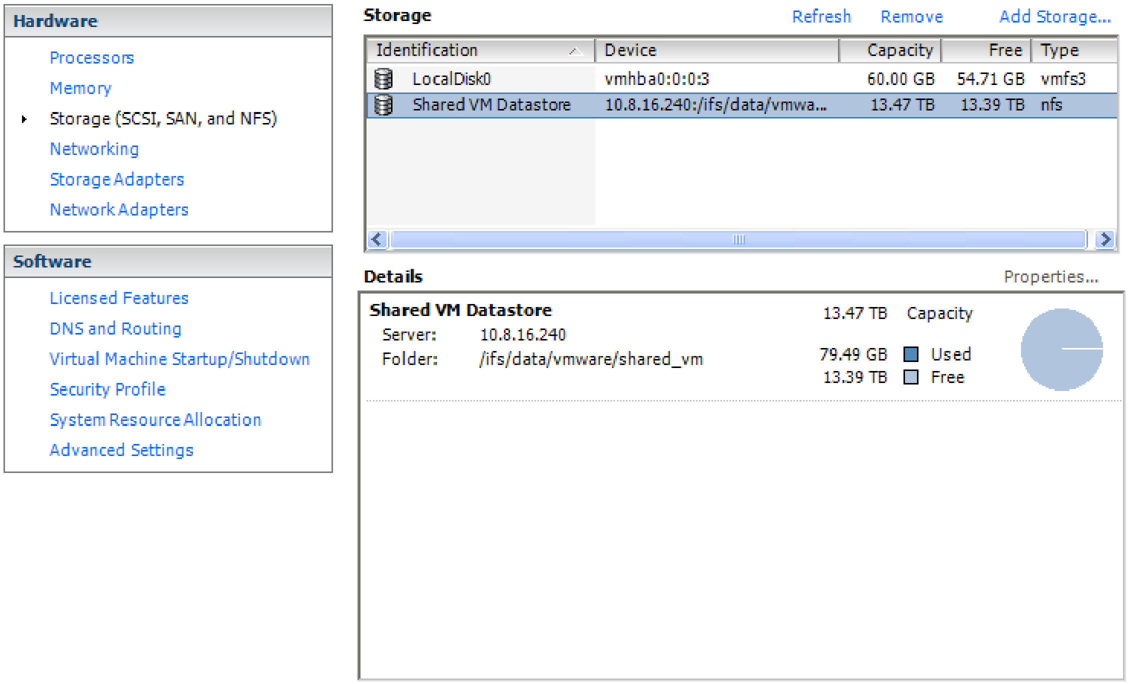
Note: If ESX load balancing option is set to **Route based on IP hash** the switch load balancing method should also be set to IP hash for aggregated port groups on both the storage and the ESX host switch ports.

Creating ESX NFS Datastores

Once the networking between the ESX server machine and the Isilon cluster has been configured, the next step is to define an ESX datastore which resides on the cluster.

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Hardware**, select **Storage** (SCSI, SAN and NFS).
2. Select **Add Storage** in the upper right of the Storage screen.
3. In the **Add Storage** wizard, select **Network File System** as the **Storage Type**. Click **Next**.
4. On the **Locate Network File System** screen, in the **Server** text field, enter the IP address of a cluster node.
5. In the **Folder** text field, enter the path to the directory representing the datastore, e.g. /ifs/data/vmware. The directory must already exist.
6. Provide a meaningful name to the datastore. The name must be unique within this ESX installation.
7. On the **Summary** screen, review the settings, and if correct, click **Finish**.

8. The new datastore will appear on the list of storage devices as type ‘**nfs’**, as illustrated in Figure 18.



Identification	Device	Capacity	Free	Type
LocalDisk0	vmhba0:0:0:3	60.00 GB	54.71 GB	vmfs3
Shared VM Datastore	10.8.16.240:/ifs/data/vmwa...	13.47 TB	13.39 TB	nfs

Shared VM Datastore
Server: 10.8.16.240
Folder: /ifs/data/vmware/shared_vm

13.47 TB Capacity
79.49 GB ■ Used
13.39 TB □ Free

Figure 18: Isilon NFS datastore configured

Enabling ESX NFS Client Access through ESX Firewall

vCenter Server, ESX hosts, and other network services are accessed using predetermined TCP and UDP ports. If you require network services such as NFS and iSCSI from outside a firewall, you may be required to reconfigure the firewall to allow access on the appropriate ports. If you are unable to access the NFS server on your network, ensure firewalls do not block the connection to the NFS server and use the vSphere client to:

1. Ensure that you can ping and vmkping the NFS server address.
2. Try to restore the mount with the command:
   ```
   esxcfg-nas –r
   ```
3. List the current mount with the command:
   ```
   esxcfg-nas –l
   ```
4. Try to remount with the command:
   ```
   esxcfg-nas -a <datastore name>  -o <nfs server hostname/ip> -s <mount point>
   ```
5. For security reasons RPC protocol filtering is enabled and the connection is refused. Open firewall ports on the network for RPC protocol and check the physical switch for any RPC protocol filtering.
6. Check ESX firewall open ports with the command:
   ```
   esxcfg-firewall -q
   ```

   Note: Check the output for NFSClient. If it is not listed, proceed to step 7.
7. Open the NFS client firewall ports 111 and 2049 on UDP and TCP protocol either in vSphere client UI or with the command:
   ```
   esxcfg-firewall --enableService nfsClient
   ```

8. Alternatively, modify the security policy in vSphere Client. Click the **Configuration tab > Security Profile > Properties**, select **NFS Client** and click **OK**.

Increasing Performance with Multiple NFS Datastores

Every NFS datastore mounted by ESX host uses two TCP connections – one for NFS control information, and the other for NFS data flow - the vast majority of the traffic to a single NFS datastore will use a single TCP connection. As a result, the upper limit throughput achievable for a single datastore (regardless of link aggregation) will be bound to a single link for traffic to that datastore.

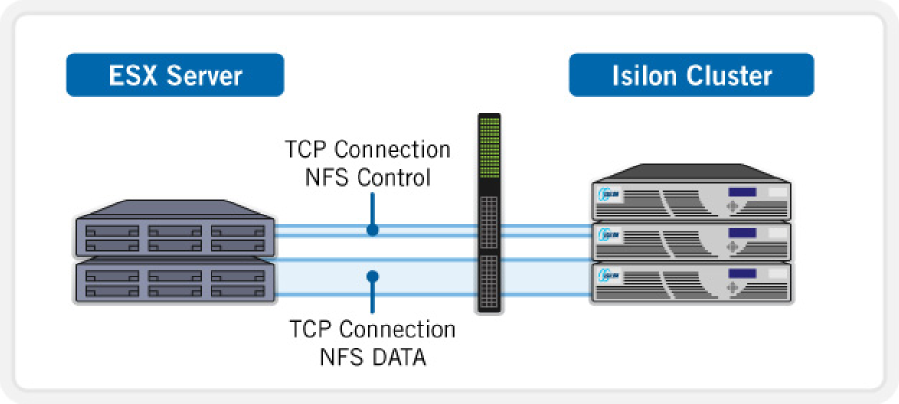


Figure 19: Single NFS datastore TCP/IP connections

All virtual machines assigned to a single datastore share this NFS mount and TCP connections. Over Gigabit Ethernet connections, the effective maximum aggregate bandwidth of all virtual machines within a single datastore is 80MB/s reads and 80MB/s writes (or 160MB/s combined). This can be translated to a maximum of 4,000 IOPs (32K IO size).

Link aggregation can increase aggregate throughput to the storage by using multiple network interfaces via a load distribution mechanism, called path selection. Path selection works on pairs of source/destination IP addresses so ESX would route traffic from multiple network interfaces to multiple datastores, pointing to different IP addresses on the Isilon cluster.

To achieve higher aggregate IO you can create multiple datastores and point each one to a separate IP address and network interface on the Isilon cluster. While this does increase the number of datastores that need to be managed, any VM can belong to any datastore because they all share a single NFS export and Isilon file system. When virtual machines are reassigned to different datastores, data does not need to move on the Isilon cluster. A VM can simply unregister from one datastore and added to the inventory of another datastore.

Isilon dynamic IPs further simplify configuration of multiple datastores because the administrator does not need to know which node and network interface the datastore is mounted to. Since dynamic IPs can be rebalanced at any given moment across the member nodes on the Isilon cluster, NFS datastore connections can be distributed “on the fly”. When a datastore connection changes from one node to another (via dynamic IP rebalancing) all disk I/O of virtual machines assigned to that datastore will be rerouted to the new node. It is recommended to increase the number of datastores to provide better load distribution across the nodes in the cluster.

For example, if there are 50 virtual machines hosted on an ESX server over two datastores pointing to two member nodes in a dynamic IP pool on the Isilon cluster, each datastore handles the aggregate I/O of all 25 VMs through a single network interface. If a network connection of one datastore is lost to one of the dynamic IP pool member, all traffic will automatically be rerouted through the remaining pool member in the dynamic IP. The remaining pool member will now serve the virtual disk I/O of 2 datastores with 50 virtual machines, effectively doubling the load on it.

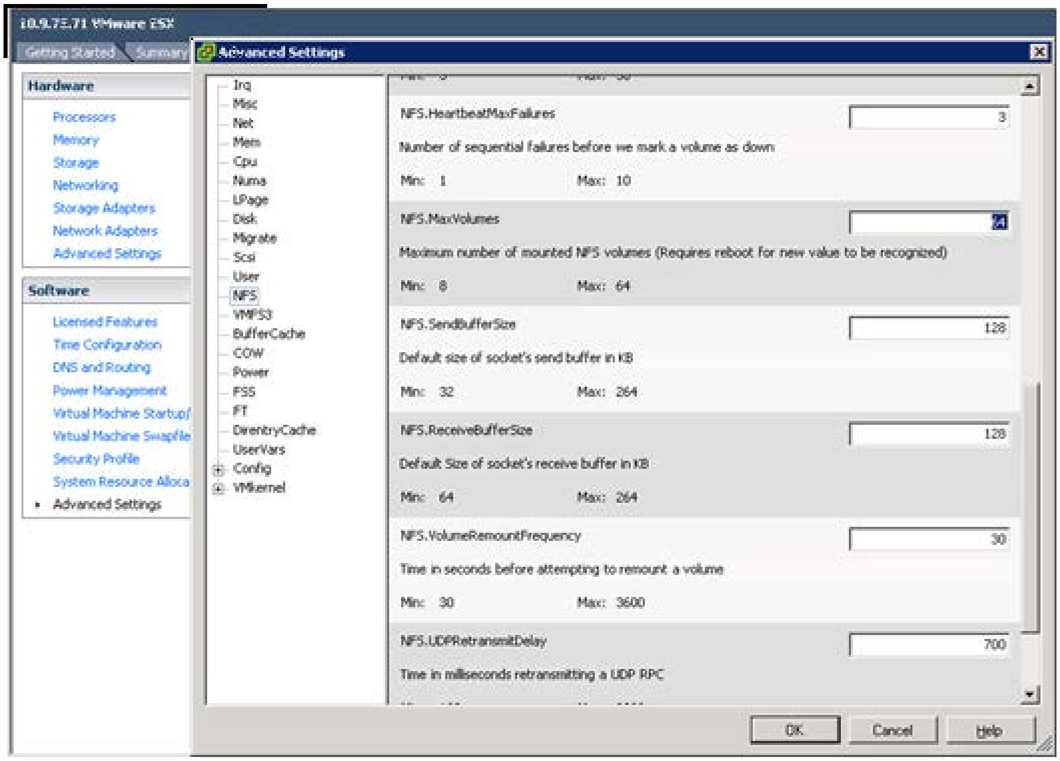
Alternatively if five datastores are connected to five dynamic IP pool members on the Isilon cluster, each datastore will only handle the aggregate I/O of 10 virtual machines through each network interface and more interfaces will be used to increase the total I/O traffic of the 50 VMs. In addition to increasing aggregate I/O, if link is lost to the Isilon cluster, the datastore connection on that link will failover to one of the 4 remaining pool members with only 10 virtual machines and overall traffic will be better balanced across the nodes in the cluster. In this degraded network state one Isilon node will handle I/O load of 2 datastores with only 20 virtual machines, and the 3 other nodes will each continue to handle one datastore I/O with10 virtual machines. Overall balance across the remaining 4 nodes will be much better than the option with only 2 datastores.

By default, the maximum number of NFS datastores in a single ESX host is limited to eight. This limit can be extended to 64 NFS datastores. To increase the maximum number of NFS datastores do the following:

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Software**, select **Advanced Settings** and select **NFS**
2. Locate the setting NFS.MaxVolumes and change the setting **64**.
3. Click **OK**.

In addition to increasing the maximum number of NFS datastores Isilon also recommends using 128K NFS send/receive buffer size. To change the NFS send/receive buffer size, complete the following:

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Software**, select **Advanced Settings** and select **NFS**
2. Locate the setting **NFS.SendBufferSize** and change to **128.**
3. Locate the setting **NFS.RecieveBufferSize** and change to **128.**
4. Click **OK**.



NFS.HeartbeatMaxFailures 3

Number of sequential failures before we mark a volume as down

Min: 1 Max: 10

NFS.MaxVolumes 64

Maximum number of mounted NFS volumes (Requires reboot for new value to be recognized)

Min: 8 Max: 64

NFS.SendBufferSize 128

Default size of socket's send buffer in KB

Min: 32 Max: 264

NFS.ReceiveBufferSize 128

Default Size of socket's receive buffer in KB

Min: 64 Max: 264

NFS.VolumeRemountFrequency 30

Time in seconds before attempting to remount a volume

Min: 30 Max: 3600

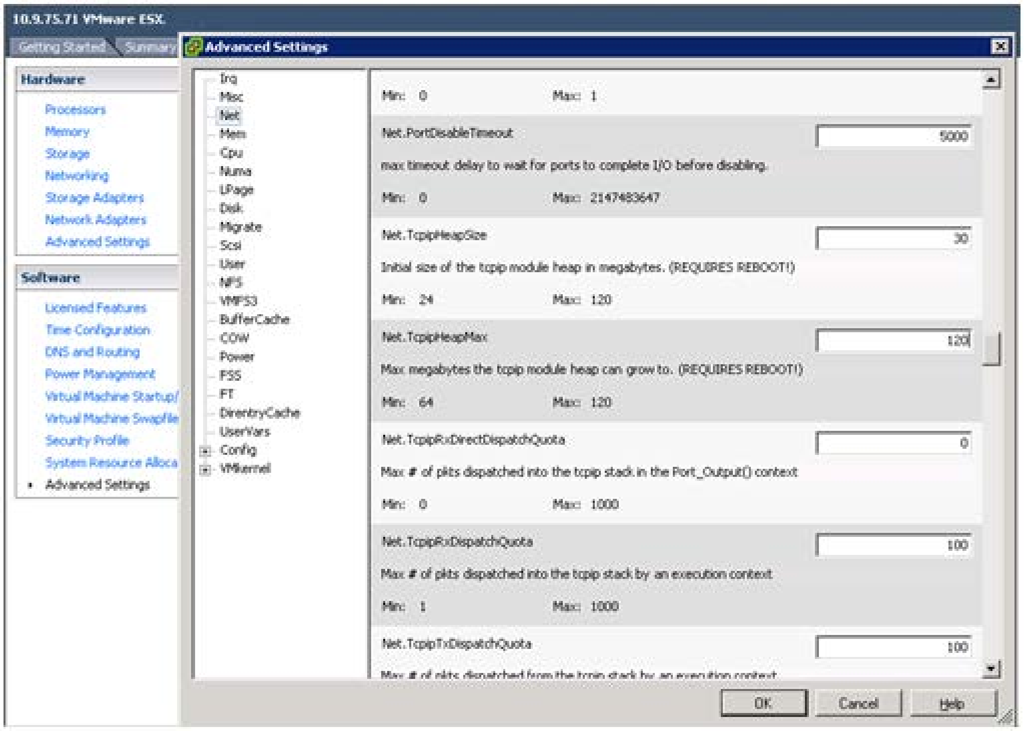
NFS.UDPRetransmitDelay 700

Time in milliseconds retransmitting a UDP RPC

Figure 20: Increasing maximum number of NFS datastores

Since more NFS datastores translates to additional NFS mounts and TCP/IP connections, the amount of memory buffer allocated by the VMkernel network stack needs to be increased. To increase the initial and maximum memory heap size for VMkernel network stack, complete the following:

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Software**, select **Advanced Settings** and select **Net**
2. Locate the **setting Net.TcpIpHeapSize** and change to **30.**
3. Locate the **setting Net.TcpIpHeapMax** and change to **120.**
4. Click **OK**.



Min: 0 Max: 1

Net.PortDisableTimeout 5000

max timeout delay to wait for ports to complete I/O before disabling.

Min: 0 Max: 2147483647

Net.TcpipHeapSize 30

Initial size of the tcpip module heap in megabytes. (REQUIRES REBOOT!)

Min: 24 Max: 120

Net.TcpipHeapMax 120

Max megabytes the tcpip module heap can grow to. (REQUIRES REBOOT!)

Min: 64 Max: 120

Net.TcpipRxDirectDispatchQuota 0

Max # of pkts dispatched into the tcpip stack in the Port_Output() context

Min: 0 Max: 1000

Net.TcpipRxDispatchQuota 100

Max # of pkts dispatched into the tcpip stack by an execution context

Min: 1 Max: 1000

Net.TcpipTxDispatchQuota 100

Max # of pkts dispatched from the tcpip stack by an execution context

Figure 21: Increasing VMKernel network memory heap size

Note: If you use the service console to access your ESX host, you can see the VMFS and NFS datastores as separate subdirectories in the **/vmfs/volumes** directory or by using the '`esxcfg-nas -l'` command

5. Isilon IQ Cluster Configuration for iSCSI VMFS Datastores

The implementation of the iSCSI protocol within Isilon IQ brings unified storage capability. Leveraging scale-out architecture, Isilon iSCSI solutions allows applications like VMware vSphere to benefit from the distributed architecture of the OneFS® operating system. This section reviews best practices directly related to the management of Logical Units (LUNs) with Isilon OneFS and integration with VMware vSphere.

Isilon iSCSI LUNs are constructed as files that reside within OneFS. Each iSCSI LUN is composed of eight extent files within a directory. These extent files can be uniquely laid out and protected like any other files on OneFS to achieve unique protection and performance capabilities. By default, LUN directories are created under the target that the LUN is assigned to, although LUNs may be moved or placed anywhere within the directory hierarchy for convenience (e.g. to enforce a single set of SmartQuotas or aid in SnapshotIQ and SyncIQ replication).

iSCSI initiators can access LUNs from their respective targets through any of the Isilon IQ cluster nodes providing a high level of performance through aggregate access and high reliability against both disk and node failures.

Note: OneFS iSCSI license is required to manage and use iSCSI on Isilon IQ clusters.

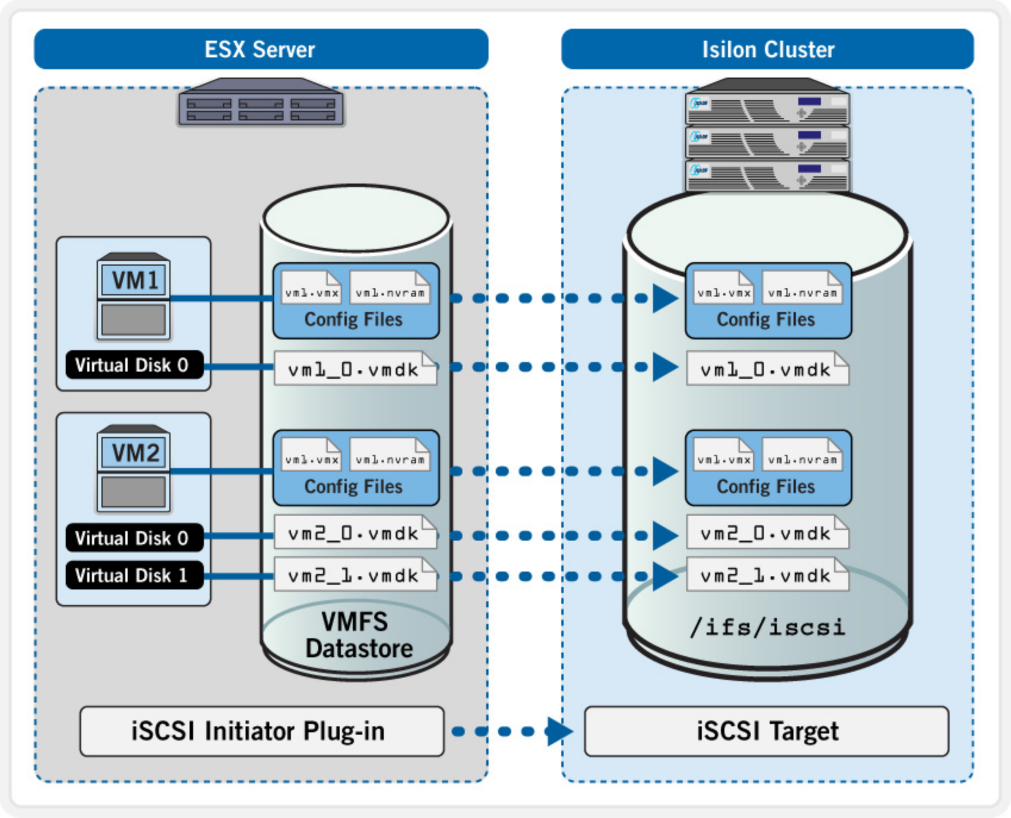


Figure 22: ESX iSCSI VMFS datastore on Isilon IQ cluster

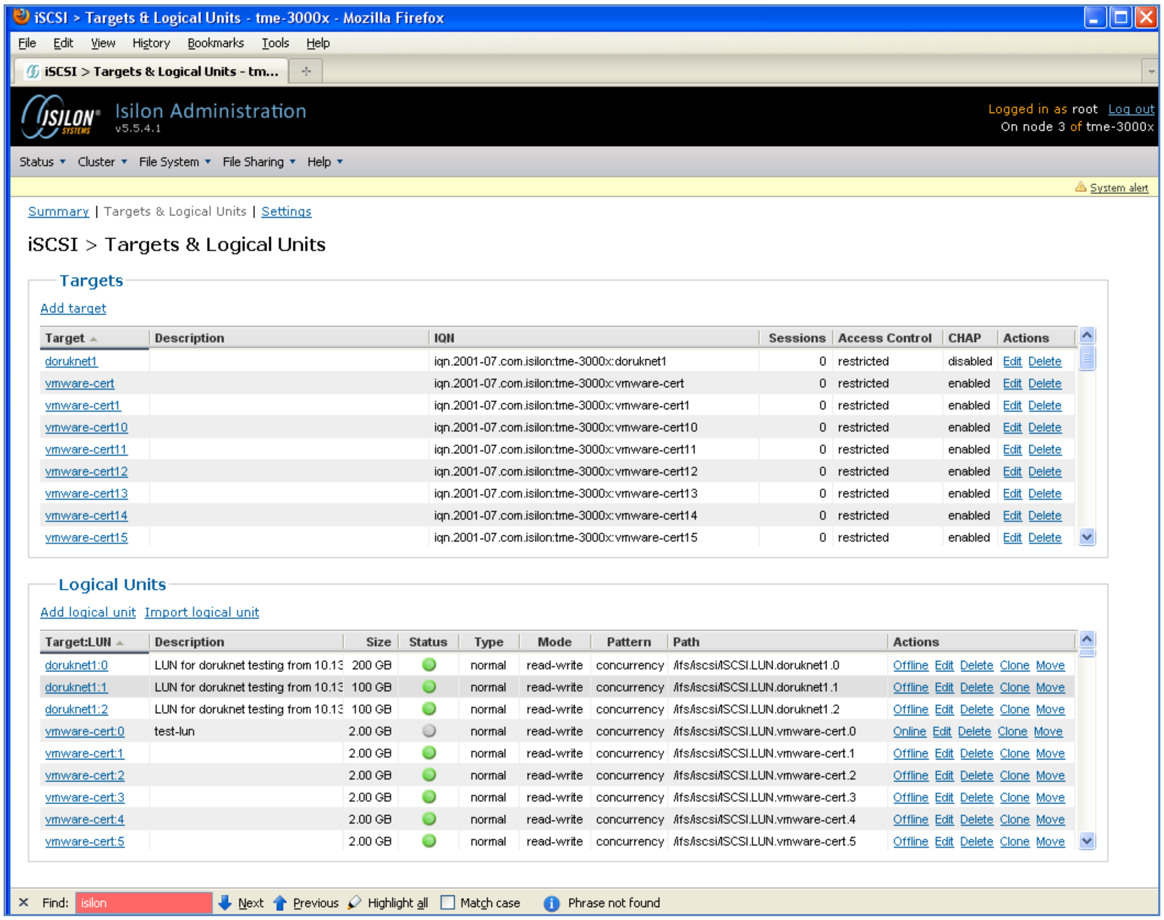
The Isilon iSCSI implementation features the following capabilities:

- Full implementation of RFCs 3720 and 5048
- Active/Passive multi-pathing supports using ESX Native Multi-Pathing (NMP).
- Support for Thin-Provisioned and Fully-Allocated LUNs
- Support for One-Way CHAP Authentication and Initiator Access Control
- Support for dynamically growing LUNs
- Support for ESX Raw Device Mapping (RDM) devices.
- Support for creating LUN clones using full normal copies, snapshot read-only copies and shadow copies writable snapshots.
- Support for VSS (Volume Shadow Service) when utilizing the Microsoft iSCSI Initiator inside virtual machines.
- The number of LUNs per target is restricted to 256. Isilon does not enforce a limit on the number of targets.

OneFS iSCSI Target Configuration

OneFS supports unlimited number of targets and 256 LUNs (numbered 0-255) within each target. Each target can be accessed through any of the cluster storage nodes IP addresses that belong to a SmartConnect static IP pool.

Each target can have its own unique CHAP authentication and target initiator access control listing.



iSCSI > Targets & Logical Units - tme-3000x - Mozilla Firefox
File Edit View History Bookmarks Tools Help
iSCSI > Targets & Logical Units - tm...
Isilon Administration v5.5.4.1
Logged in as root Log out
On node 3 of tme-3000x
Status ▾ Cluster ▾ File System ▾ File Sharing ▾ Help ▾
System alert
Summary | Targets & Logical Units | Settings

iSCSI > Targets & Logical Units

Targets

Add target

Target	Description	IQN	Sessions	Access Control	CHAP	Actions
doruknet1		iqn.2001-07.com.isilon:tme-3000x:doruknet1	0	restricted	disabled	Edit Delete
vmware-cert		iqn.2001-07.com.isilon:tme-3000x:vmware-cert	0	restricted	enabled	Edit Delete
vmware-cert1		iqn.2001-07.com.isilon:tme-3000x:vmware-cert1	0	restricted	enabled	Edit Delete
vmware-cert10		iqn.2001-07.com.isilon:tme-3000x:vmware-cert10	0	restricted	enabled	Edit Delete
vmware-cert11		iqn.2001-07.com.isilon:tme-3000x:vmware-cert11	0	restricted	enabled	Edit Delete
vmware-cert12		iqn.2001-07.com.isilon:tme-3000x:vmware-cert12	0	restricted	enabled	Edit Delete
vmware-cert13		iqn.2001-07.com.isilon:tme-3000x:vmware-cert13	0	restricted	enabled	Edit Delete
vmware-cert14		iqn.2001-07.com.isilon:tme-3000x:vmware-cert14	0	restricted	enabled	Edit Delete
vmware-cert15		iqn.2001-07.com.isilon:tme-3000x:vmware-cert15	0	restricted	enabled	Edit Delete

Logical Units

Add logical unit Import logical unit

Target:LUN	Description	Size	Status	Type	Mode	Pattern	Path	Actions
doruknet1:0	LUN for doruknet testing from 10.13	200 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.doruknet1.0	Offline Edit Delete Clone Move
doruknet1:1	LUN for doruknet testing from 10.13	100 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.doruknet1.1	Offline Edit Delete Clone Move
doruknet1:2	LUN for doruknet testing from 10.13	100 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.doruknet1.2	Offline Edit Delete Clone Move
vmware-cert:0	test-lun	2.00 GB	○	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.0	Online Edit Delete Clone Move
vmware-cert:1		2.00 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.1	Offline Edit Delete Clone Move
vmware-cert:2		2.00 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.2	Offline Edit Delete Clone Move
vmware-cert:3		2.00 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.3	Offline Edit Delete Clone Move
vmware-cert:4		2.00 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.4	Offline Edit Delete Clone Move
vmware-cert:5		2.00 GB	●	normal	read-write	concurrency	/ifs/iscsi/ISCSI.LUN.vmware-cert.5	Offline Edit Delete Clone Move

Figure 23: iSCSI Targets and LUNs on OneFS

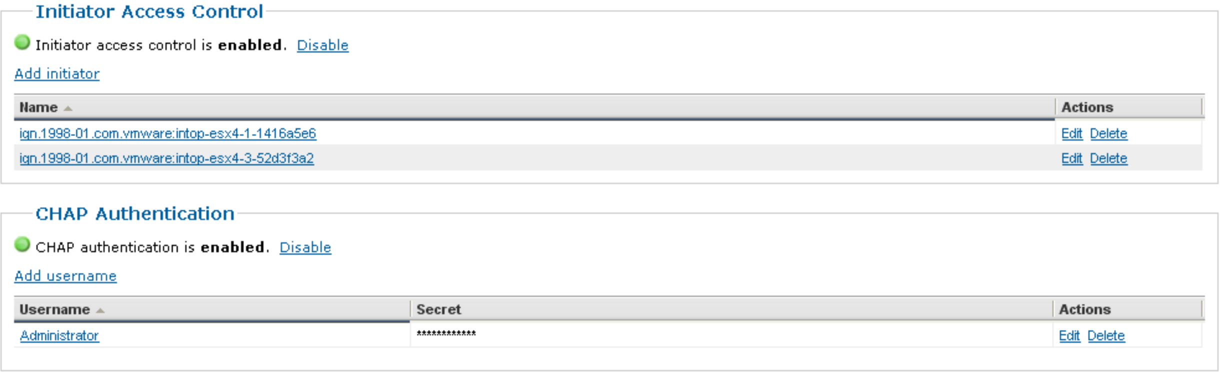
Target CHAP Authentication

Isilon® recommends the use of CHAP for security. Isilon supports per-target one-way CHAP authentication using username/password pairs. Usernames follow the same rules as target names (arbitrary strings following standard hostname rules -- colons and periods are okay).

Target Initiator Access Control

Isilon supports per-target access control. Targets can be in one of two states: Open to all initiators, or closed to all initiators except for those on the allowed initiator list. The initiator list can be empty, leaving three possible configurations: 1. Open to all, 2. closed to all or, 3. open to a specified few. If a target is closed to a given initiator, that target will not be revealed to that initiator upon a **SendTargets** discovery, and access will be denied to that initiator upon any attempt to log in or scan for LUNs.

Note: Isilon recommends restricting access to LUNs used for VMFS datastores by listing the ESX iSCSI initiators that share VMFS volumes in a virtual environment. Taking this action ensures no other iSCSI initiators can access and corrupt the VMFS datastores.



Initiator Access Control

● Initiator access control is **enabled**. Disable

Add initiator

Name	Actions
iqn.1998-01.com.vmware:intop-esx4-1-1416a5e6	Edit Delete
iqn.1998-01.com.vmware:intop-esx4-3-52d3f3a2	Edit Delete

CHAP Authentication

● CHAP authentication is **enabled**. Disable

Add username

Username	Secret	Actions
Administrator	************	Edit Delete

Figure 24: Setting Target CHAP and Initiator Access Control

OneFS iSCSI LUN Configuration

LUN Type

Isilon supports provisioning LUNs as **Thin** or **Thick**. Provisioning a LUN as thin will take less time at creation and by definition will not claim any space until blocks are written by the ESX host. Thinly provisioned LUNs may initially have higher performance than thickly provisioned LUNs as every initial write in a thinly provisioned LUN does not result in a read-modify-write operation within the OneFS file system (unless LUNs are mirrored). Of course, this improvement goes away once a block is being modified within the thin LUN. Every rewrite within a thin LUN could potentially result in the same read-modify-write operation as a thickly provisioned LUN.

OneFS LUN settings for Optimizing ESX iSCSI VMFS I/O Performance

LUN layout settings determine how LUNs are protected and striped across disks on OneFS. Layout settings affect performance and utilization and should be set according to the type of I/O load generated by the ESX host and its underlying virtual machines. In most cases I/O load generated by a set of virtual machines in an ESX host is mixed read/write and relatively random in nature. However, in some cases I/O load can be dominated by one type of access or another. Below are LUN layout recommendations for specific types of I/O intensive load:

1. Very random writes within a large LUN have been shown to benefit from setting the **Access Pattern** to **Streaming**. With our current testing, **Concurrency** versus **Streaming** has demonstrated no performance increase or decrease except for this one case.
2. In most cases, **2x mirroring** protection will result in better performance since parity reads aren't required during the write process. However, if your workflow is primarily reads, 2x mirroring won't provide as much benefit. If space is a primary concern then you may be willing to sacrifice some write performance to avoid mirroring space overhead.
3. **Write Caching** on Isilon iSCSI LUNs is turned off by default. Turning on **Write Caching** can result in write performance improvements, but there is a risk of corruption if a node loses power or crashes while uncommitted data is in the write cache. One exception would be to turn write caching on during the creation of a thickly allocated LUN, and turning write caching off once the LUN creation is completed.
4. Disable OneFS **read prefetch** if many virtual machines require high ratio of small random read operations. Disabling **read prefetch** instructs OneFS to avoid prefetching adjacent file blocks to eliminate prefetch latency overhead. Read prefetch is only valuable for sequential read workflows.

To create a new LUN or edit an existing LUN's settings:

1. From the WebUI select **File System→iSCSI→Targets & Logical Units**
2. In the **Logical Units** box select **Add logical unit**

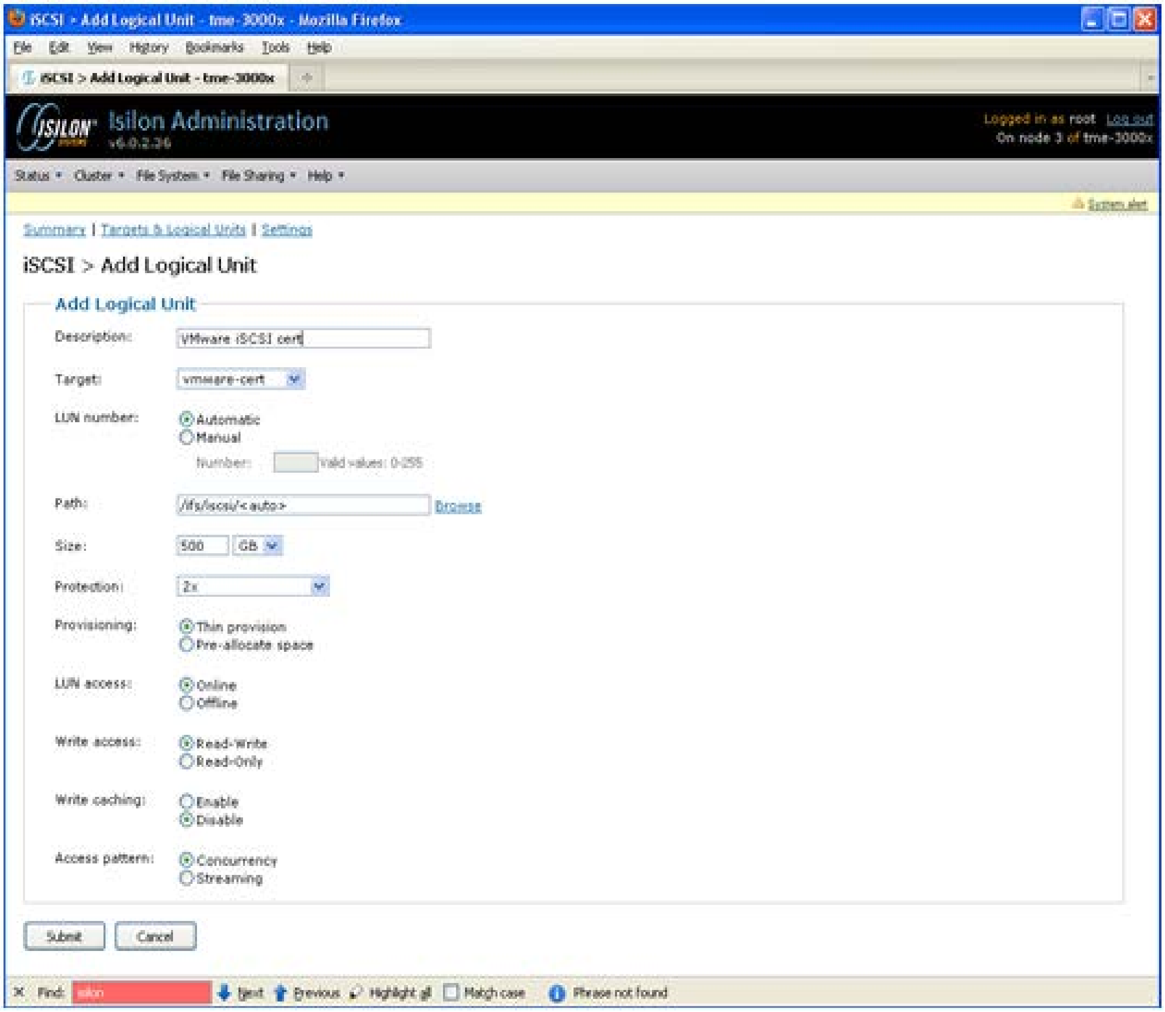
3. Select the Target associated with this logical unit and add a description.
4. Set the LUN **Provisioning**, **Protection** and **Access Pattern** based on the guidelines provided above.

To disable cluster read prefetch setting:

1. Log on to any of the nodes in the cluster over SSH connection
2. At the login prompt issue the following command: `sysctl efs.bam.enable_prefetch=1`
3. To make this setting persistent across cluster reboots add the following line to the file /etc/mcp/override/sysctl.conf: `efs.bam.enable_prefetch=1`

To change LUN protection mode

Before setting **2X mirroring** protection on a LUN you must changed the **FlexProtect** global setting to **Advanced**. From the WebUI select **File System→File System Settings->Flex Protect Policy** and set to **Advanced.** This can be done through the Web based administration User Interface or using the following CLI command: **`‘isi flexprotect advanced’`**


iSCSI > Add Logical Unit - tme-3000x - Mozilla Firefox
File Edit View History Bookmarks Tools Help
iSCSI > Add Logical Unit - tme-3000x
Isilon Administration
v6.0.2.36
Logged in as root Log out
On node 3 of tme-3000x
Status ▾ Cluster ▾ File System ▾ File Sharing ▾ Help ▾
System alert
Summary | Targets & Logical Units | Settings
Submit
Cancel
Find: isilon Next Previous Highlight all □ Match case Phrase not found

iSCSI > Add Logical Unit

Add Logical Unit

Description: VMware iSCSI cert

Target: vmware-cert

LUN number: ● Automatic ○ Manual
Number: Valid values: 0-255

Path: /ifs/iscsi/<auto> Browse

Size: 500 GB

Protection: 2x

Provisioning: ● Thin provision ○ Pre-allocate space

LUN access: ● Online ○ Offline

Write access: ● Read-Write ○ Read-Only

Write caching: ○ Enable ● Disable

Access pattern: ● Concurrency ○ Streaming

Figure 25: Creating or editing a LUN

OneFS Networking Best Practices for iSCSI

Since iSCSI and NFS protocols both work over Ethernet/IP infrastructure they share some common network design guidelines and best practices. Those common best practices are described in section General Networking Guidelines for Ethernet/IP based ESX Datastores above. But NFS and iSCSI have different mechanisms when using multiple network paths to achieve network redundancy and increase aggregate network throughput.

ESX Mutli-pathing for iSCSI

Although technically possible to configure, VMware does not recommend using network link aggregation techniques for iSCSI network redundancy for vSphere. VMware recommends using iSCSI built-in multi-pathing (MPIO) which defines end-to-end paths from iSCSI initiator to target. This is because, unlike NFS, block storage architecture uses MPIO provided in the storage stack and not the networking stack for multi-pathing behavior.

Note: ESX 3.5 and below still use link aggregation, as described in the NFS network guidelines, for iSCSI network redundancy because iSCSI MPIO support has only been introduced in vSphere (ESX 4.0).

iSCSI MPIO relies on the ability of the iSCSI initiator to discover multiple paths to storage targets and establish a path from each of the VMKernel ports that are bound to the iSCSI initiator and each of the target portals defined by a unique target IP and port.

To make sure the VMKernel ports used by the iSCSI initiator are actual paths to storage, ESX configuration requires that they are connected to a portgroup that only has one active uplink interface and no standby ones. This way, if the interface is unavailable, the storage path is down and the iSCSI multi-pathing mechanism can choose a different path.

On the Isilon storage end iSCSI targets can be represented by IP addresses assigned to a link aggregation trunk on individual nodes for ESX 3.5 and below, or IP addresses assigned to individual interfaces across all nodes in the cluster.

Isilon recommends the following steps to maximize performance and availability for ESX iSCSI datastores:

- Dedicate a static IP pool on the Isilon cluster for managing iSCSI target IPs. Isilon does not support dynamic IP addresses for iSCSI targets and will not publish targets on dynamic IP addresses in SendTarget discovery response.
- Create multiple VMKernel port groups on the ESX host, with a single active network interface and no standby interfaces. Use **iSCSI port binding** to associate those VMkernel port groups with the iSCSI initiator.
- Make sure the selected Storage Array Type Plugin (SATP) is **Active/Active** and the Path Selection Policy (PSP) is **FIXED**. If those are not set by default they can be manually changed on a per target, datastore or LUN level.
- To avoid lock-contention as a result of multiple ESX hosts accessing the same LUN from different target nodes, make sure all ESX hosts use the same preferred path to the same LUN.

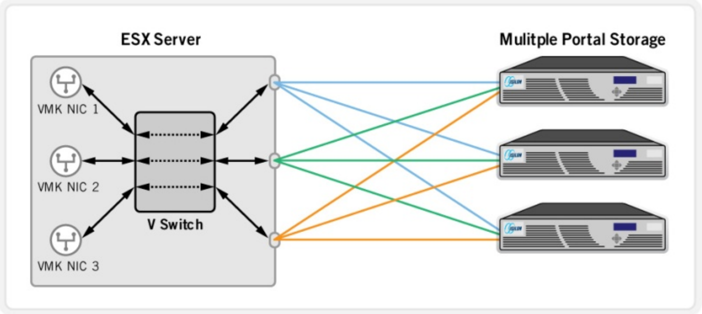


Figure 26: Multi-path using iSCSI port binding in vSphere

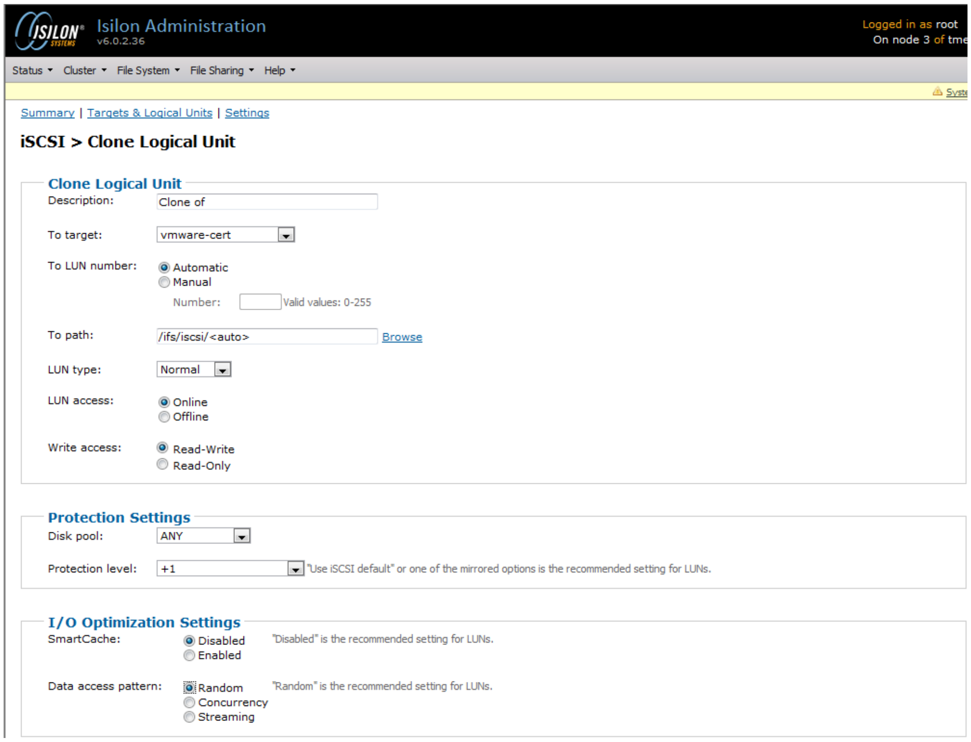
Cloning LUNs for ESX Datastores

With vSphere client, VMware allows iSCSI VMFS datastores to be provisioned with thin-provisioned VMDKs. Isilon offers further improvements in storage utilization and faster provisioning by supporting thinly provisioning new iSCSI LUNs (described above) and creating **Shadow** clones of existing iSCSI LUNs on the storage cluster. **Shadow** cloned LUNs act as space efficient writeable copies of iSCSI LUNs. They are created instantaneously by selection the **“Shadow”** type when cloning a LUN and only consume space on the Isilon file system when an iSCSI initiator modifies the clone LUN data. A **Shadow** cloned LUN can be used to store different versions of gold images or templates of virtual machines by cloning an entire VMFS datastore, or to clone RDM LUNs associated with individual VMs.

Note: prior to creating shadow clones, it might be necessary to shutdown all VMs in the datastore stored on the LUN. Otherwise, the VM data in the LUN may be inconsistent at the time it is cloned. Snapshot consistency is covered in more detail in **Chapter 9 Working with Snapshots.**

To create a new LUN as a Shadow clone:

1. From the WebUI select **File System→iSCSI→Targets & Logical Units**
2. Select a LUN from the list of Logical Units and click on **Clone**
3. Change **Description** and other settings of the LUN clone.
4. Click **Submit**



iSCSI > Clone Logical Unit

Clone Logical Unit

Description: Clone of

To target: vmware-cert

To LUN number: ● Automatic ○ Manual
Number: Valid values: 0-255

To path: /ifs/iscsi/<auto> Browse

LUN type: Normal

LUN access: ● Online ○ Offline

Write access: ● Read-Write ○ Read-Only

Protection Settings

Disk pool: ANY

Protection level: +1 "Use iSCSI default" or one of the mirrored options is the recommended setting for LUNs.

I/O Optimization Settings

SmartCache: ● Disabled ○ Enabled "Disabled" is the recommended setting for LUNs.

Data access pattern: ● Random ○ Concurrency ○ Streaming "Random" is the recommended setting for LUNs.

Figure 27: Creating a shadow cloned LUN

The cloned datastore can be resigned and new virtual machines (that are clones of the original ones) can be powered on. However, these limitations need to be taken into account when using shadow clones:

- Shadow cloned LUNs are linked to the original base LUN they were cloned from but changes to setting on the cloned LUN do not affect the base LUN. Since any blocks changed on the cloned LUN are written to a new location on the clone LUN extents protection and LUN layout settings are only applied to the modified blocks.

- The only LUNs that can be fully backed-up, restored or replicated using Isilon SyncIQ from within an Isilon IQ cluster are normal LUNs and normal clones. This is because cluster based backup, restore and replication applications do not have knowledge of the base LUN and snapshot info associated with them. For this reason, it is recommended that backups be performed at the virtual machine level or using VMware array data protection (VADP discussed later).

Note: iSCSI shadow clones, as well as other types of clones, requires that Isilon SnapshotIQ® be licensed.

When a cloned LUN contains a VMFS datastore copy you now have two datastores with the same VMFS UUID signature. You can now choose whether to mount the LUN copy with the original VMFS signature or create a new VMFS signature. Since you cannot have two VMFS with the same signature mounted at the same time, you will have to unmount the original VMFS before mounting the cloned VMFS with the same signature. Otherwise you can choose to create a new signature to the new clone while keeping the original one mounted.

Note: Datastore "re-signaturing" is irreversible. Once you perform a datastore resignature on a LUN clone it is no longer treated as a copy, but rather a unique LUN with a unique VMFS datastore.

6. ESX Configuration for iSCSI VMFS Datastores

Much of the information in this section is paraphrased from the VMware **iSCSI SAN Configuration Guide** available online from VMware. We've included a selection of key considerations and configuration guidelines but encourage the readers to review the VMware Best Practice Guide for a more comprehensive review of iSCSI setup and advanced settings.

LUN Sizing and Setting Considerations

A single iSCSI LUN can only belong to one VMFS datastore. VMware can create a single VMFS datastore from multiple iSCSI LUNs as extents to the VMFS volume. Each LUN should have the correct protection and layout settings for applications in virtual machines in that LUN. If multiple virtual machines access the same VMFS, use VMDK disk shares to prioritize virtual machines IO access.

The maximum VMware supported LUN size is 2TB, however larger datastores can be created through VMFS LUN spanning. VMFS spanning leverages VMFS extents to link together multiple LUNs into a single datastore. Because LUN size consideration must take into account the I/O load that virtual machines in that datastore generate, a commonly deployed size for a VMFS datastore is between 300GB and 700GB and 5-10 virtual machines. In situation where a single datastore is overloaded, VMware provides Storage vMotion as a means to redistribute VM storage to alternative datastores without disruption to the VM.

Isilon recommends using more, smaller VMFS datastores because **t**here is less wasted storage space (thick provisioned LUNs). You also can tune each underlying LUN in a more granular method to meet a specific set of application IO and data protection requirements for a virtual machine. As more datastores are used concurrently the aggregate IO increases, because more paths can be used in parallel and contention on a per VMFS volumes is reduced.

However, there are some benefits to using less, larger VMFS datastores including more flexibility in creating virtual machines or resizing virtual disks without asking for storage space and allowing you to manage fewer VMFS datastores.

ESX Basic Network Configuration for iSCSI Datastores

Because like NFS, iSCSI requires network connectivity to access data stored on remote servers, before enabling the iSCSI initiator, you must first configure VMkernel networking. The steps to setting up iSCSI networking and enabling iSCSI access to targets include:

1. Create a **vSwitch** and a **VMkernel** port for physical network interfaces.
2. Enable the **software iSCSI initiator**.
3. If you use multiple network adapters, setup **multi-pathing** on the ESX host using the **port binding** technique. With port binding, you create a separate VMkernel port for each physical network interface using 1:1 mapping either in the same vSwitch or a separate vSwitch for each VMKernel.

Creating a Virtual Switch (vSwitch)

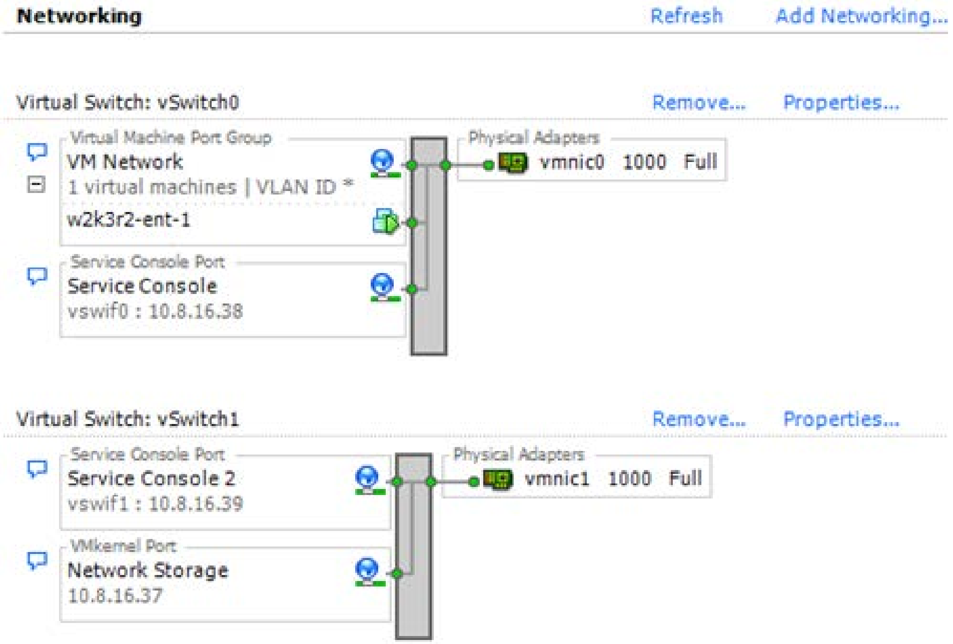
The first step is to create a virtual switch for all network traffic between the ESX server machine and the Isilon cluster.

1. In the vSphere Client, select the ESX server machine in the left-side tree view, then select the **Configuration** tab in the right-side pane.
2. Under **Hardware**, select **Networking**, then select **Add Networking**.
3. In the **Add Network Wizard**, in the **Connection Types** section, select **VMkernel**, and click **Next**.
4. On the **Network Access** screen, select **Create a virtual switch**, or select an existing virtual switch. Click **Next**. Isilon recommends using at least one dedicated network interface card (NIC) for network storage traffic. This will ensure good performance, as well as isolate any problems from other traffic.
5. On the **Connection Settings** screen, enter a network label and optional **VLAN ID**. It's often helpful to give the virtual switch a meaningful label, such as **"iSCSI Storage".**
6. In the **IP Settings** section, enter an IP address and subnet mask for the **VMkernel port**.
7. If necessary, click the **Edit** button to change the default gateway. Click **Next** to go to the **Summary** screen.
8. On the **Summary** screen, review the settings, and if correct, click **Finish**.

Configuring a Service Console (ESX 3.5 Only)

ESX 3.5 requires you to configure a service console on the virtual switch you just created. Without a service console, it is possible for the ESX server machine to lose connectivity to storage located on the virtual switch. This step is NOT necessary for vSphere and ESX 4.1.

7. In the VI Client, on the **Configuration** tab for the ESX server machine, select **Properties** next to the virtual switch that you just created.
8. In the **Properties** dialog, on the **Ports** tab, click **Add**.
9. In the **Add Network Wizard**, in the **Connection Types** section, select **Service Console**, and click **Next**.
10. On the **Connection Settings** screen, enter a network label and optional **VLAN ID**.
11. The console can be given a static IP address or obtain one via DHCP, then click **Next**.
12. On the **Summary** screen, review the settings, and if correct, click **Finish**.



Networking

Virtual Switch: vSwitch0

Virtual Machine Port Group
VM Network
1 virtual machines | VLAN ID *
w2k3r2-ent-1

Service Console Port
Service Console
vswif0 : 10.8.16.38

Physical Adapters
vmnic0 1000 Full

Virtual Switch: vSwitch1

Service Console Port
Service Console 2
vswif1 : 10.8.16.39

VMkernel Port
Network Storage
10.8.16.37

Physical Adapters
vmnic1 1000 Full

Figure 28: Example ESX network configuration

Enabling iSCSI Initiator Firewall Settings

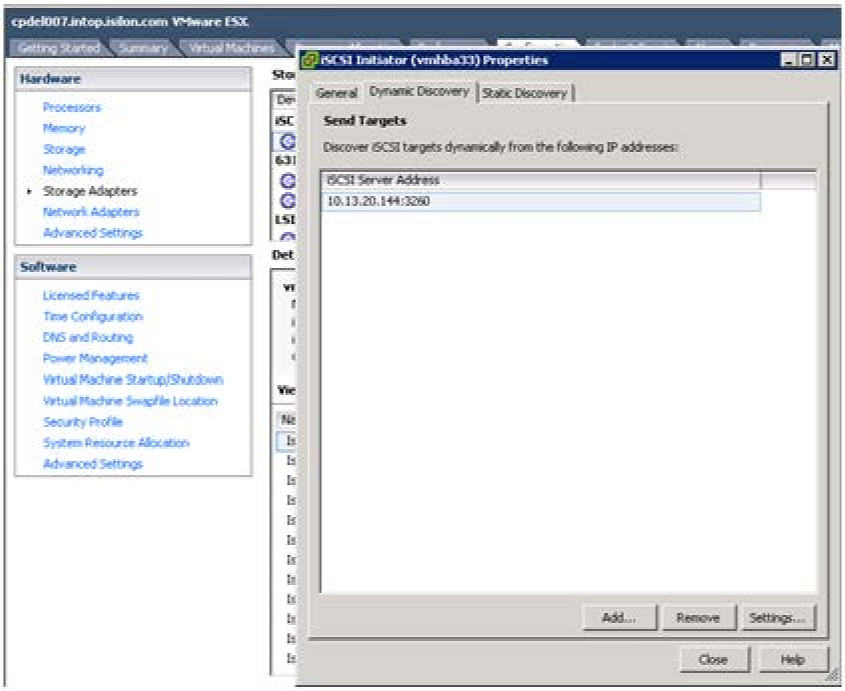
You must enable your software iSCSI initiator so that ESX/ESXi can use it to access iSCSI storage:

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Hardware**, select **Storage** (SCSI, SAN and NFS).
2. Select the iSCSI initiator to configure and click **Properties**.
3. Click **Configure**.
4. In the **General Properties** dialog box select **Enabled** to enable the initiator.
5. Click **OK** to save your changes.

Configuring Discovery Address for iSCSI Initiators

Before creating new VMFS datastores from iSCSI LUNs, the iSCSI initiator needs to discover all available LUNs on the network. This is done by listing the discover IP addresses used for two discovery methods:

- **Dynamic Discovery** using the **SendTargets** command. The iSCSI initiator sends the SendTargets request to one of the nodes in the Isilon cluster using a well known IP address. The cluster responds by supplying a list of available target IP address and ports (also known as target portals) to the initiator. The names and IP addresses of these targets appear on the **Static Discovery** tab. If you remove a static target added by dynamic discovery, the target might be returned to the list the next time a rescan happens, the HBA is reset, or the host is rebooted.
- **Static Discovery** by manually inserting target portals**.** The initiator does not have to perform any discovery. The initiator is provided with a list of target portals by the administrator.



Send Targets

Discover iSCSI targets dynamically from the following IP addresses:

iSCSI Server Address
10.13.20.144:3260

Figure 29: ESX dynamic discovery target IP address

Once a list of target portals is provided, they are combined with the list of iSCSI initiator interfaces to form multiple paths between the iSCSI initiators and the iSCSI targets in a mash topology. At that point, further multi-pathing configuration can take place. Multi-pathing management is covered further in this section.

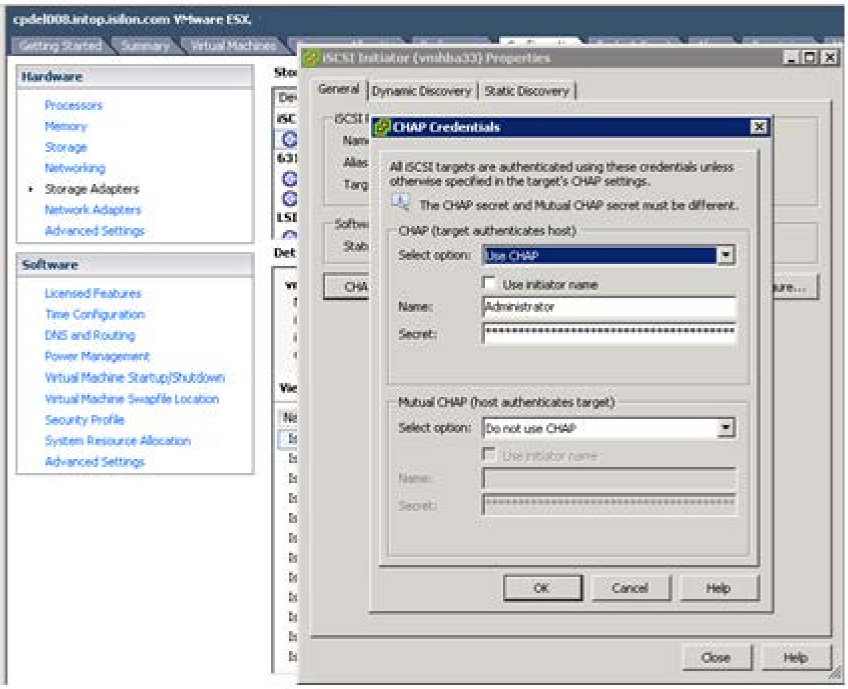
Configuring CHAP

Before configuring CHAP, check whether CHAP is enabled on the Isilon IQ cluster. If CHAP is not enabled, enable it for your initiators, making sure that the CHAP authentication credentials match the credentials on the Isilon cluster.

Note: Isilon only supports **One-way CHAP.** In one-way CHAP authentication, the target authenticates the initiator, but the initiator does not authenticate the target.

To configure the iSCSI initiator with CHAP make sure that the iSCSI target on the Isilon cluster has been configured with CHAP settings and use the following steps on the ESX host:

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Hardware**, select **Storage** (SCSI, SAN and NFS).
2. Select the iSCSI initiator to configure and click **Properties**.
3. Click **CHAP**.
4. In the **CHAP (target authenticates host)** select **Use CHAP**.
5. Enter **Name** and **Secret** the same as in the iSCSI target CHAP settings.
6. Make sure to uncheck the **Use Initiator Name** check box.
7. Make sure to select **Do not use CHAP** in the **Mutual Chap (host authenticates target)** dialog box make
8. Click **OK** to save your changes.
9. Close the iSCSI initiator **Properties** dialog box.



All iSCSI targets are authenticated using these credentials unless otherwise specified in the target's CHAP settings.

The CHAP secret and Mutual CHAP secret must be different.

CHAP (target authenticates host)

Select option: Use CHAP

□ Use initiator name

Name: Administrator

Secret: **

Mutual CHAP (host authenticates target)

Select option: Do not use CHAP

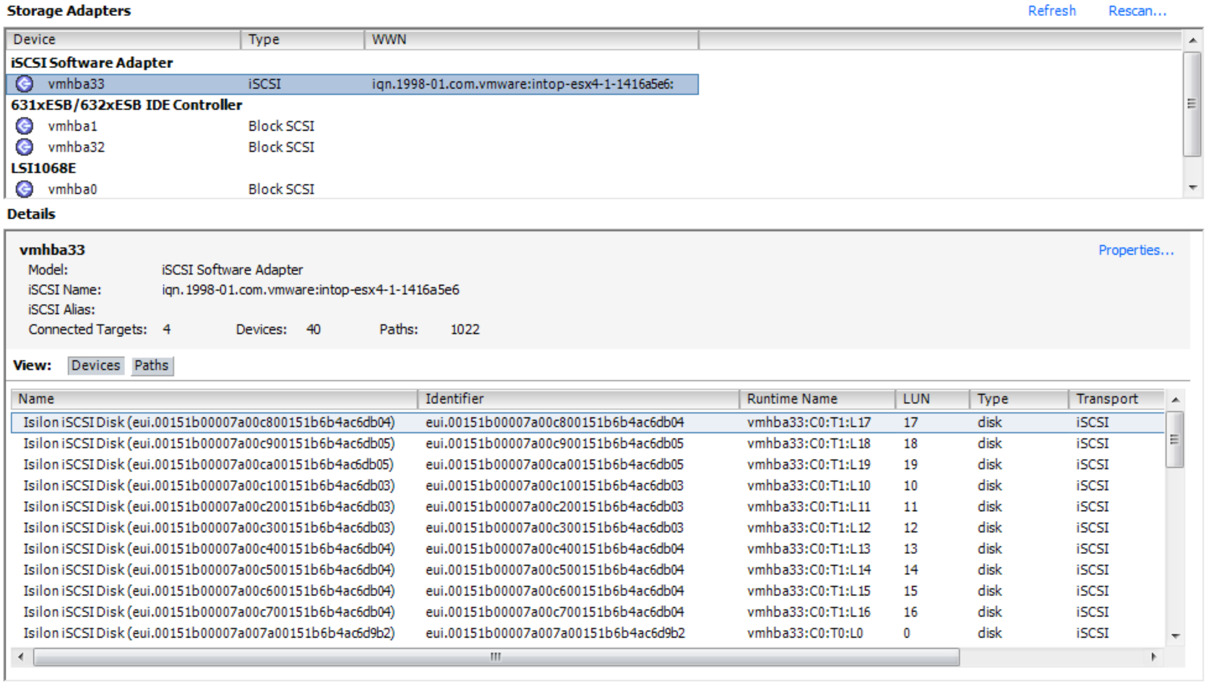
□ Use initiator name

Name:

Secret: **

Figure 30: ESX dynamic discovery target IP address

Once CHAP is configured, the iSCSI initiator can rescan all available devices on the targets that it has access to and was able to authenticate to. A rescan will occur automatically after the software iSCSI is enabled, or CHAP and dynamic, or static, discovery setting change. A rescan can also be initiated by the user either thorough vSphere client or the ESX CLI.



Storage Adapters

Device	Type	WWN
iSCSI Software Adapter		
vmhba33	iSCSI	iqn.1998-01.com.vmware:intop-esx4-1-1416a5e6:
631xESB/632xESB IDE Controller		
vmhba1	Block SCSI	
vmhba32	Block SCSI	
LSI1068E		
vmhba0	Block SCSI	

Details

vmhba33
Model: iSCSI Software Adapter
iSCSI Name: iqn.1998-01.com.vmware:intop-esx4-1-1416a5e6
iSCSI Alias:
Connected Targets: 4 Devices: 40 Paths: 1022

Name	Identifier	Runtime Name	LUN	Type	Transport
Isilon iSCSI Disk (eui.00151b00007a00c800151b6b4ac6db04)	eui.00151b00007a00c800151b6b4ac6db04	vmhba33:C0:T1:L17	17	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c900151b6b4ac6db05)	eui.00151b00007a00c900151b6b4ac6db05	vmhba33:C0:T1:L18	18	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00ca00151b6b4ac6db05)	eui.00151b00007a00ca00151b6b4ac6db05	vmhba33:C0:T1:L19	19	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c100151b6b4ac6db03)	eui.00151b00007a00c100151b6b4ac6db03	vmhba33:C0:T1:L10	10	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c200151b6b4ac6db03)	eui.00151b00007a00c200151b6b4ac6db03	vmhba33:C0:T1:L11	11	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c300151b6b4ac6db03)	eui.00151b00007a00c300151b6b4ac6db03	vmhba33:C0:T1:L12	12	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c400151b6b4ac6db04)	eui.00151b00007a00c400151b6b4ac6db04	vmhba33:C0:T1:L13	13	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c500151b6b4ac6db04)	eui.00151b00007a00c500151b6b4ac6db04	vmhba33:C0:T1:L14	14	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c600151b6b4ac6db04)	eui.00151b00007a00c600151b6b4ac6db04	vmhba33:C0:T1:L15	15	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a00c700151b6b4ac6db04)	eui.00151b00007a00c700151b6b4ac6db04	vmhba33:C0:T1:L16	16	disk	iSCSI
Isilon iSCSI Disk (eui.00151b00007a007a00151b6b4ac6d9b2)	eui.00151b00007a007a00151b6b4ac6d9b2	vmhba33:C0:T0:L0	0	disk	iSCSI

Figure 31: ESX list of iSCSI LUNs after iSCSI initiator rescan

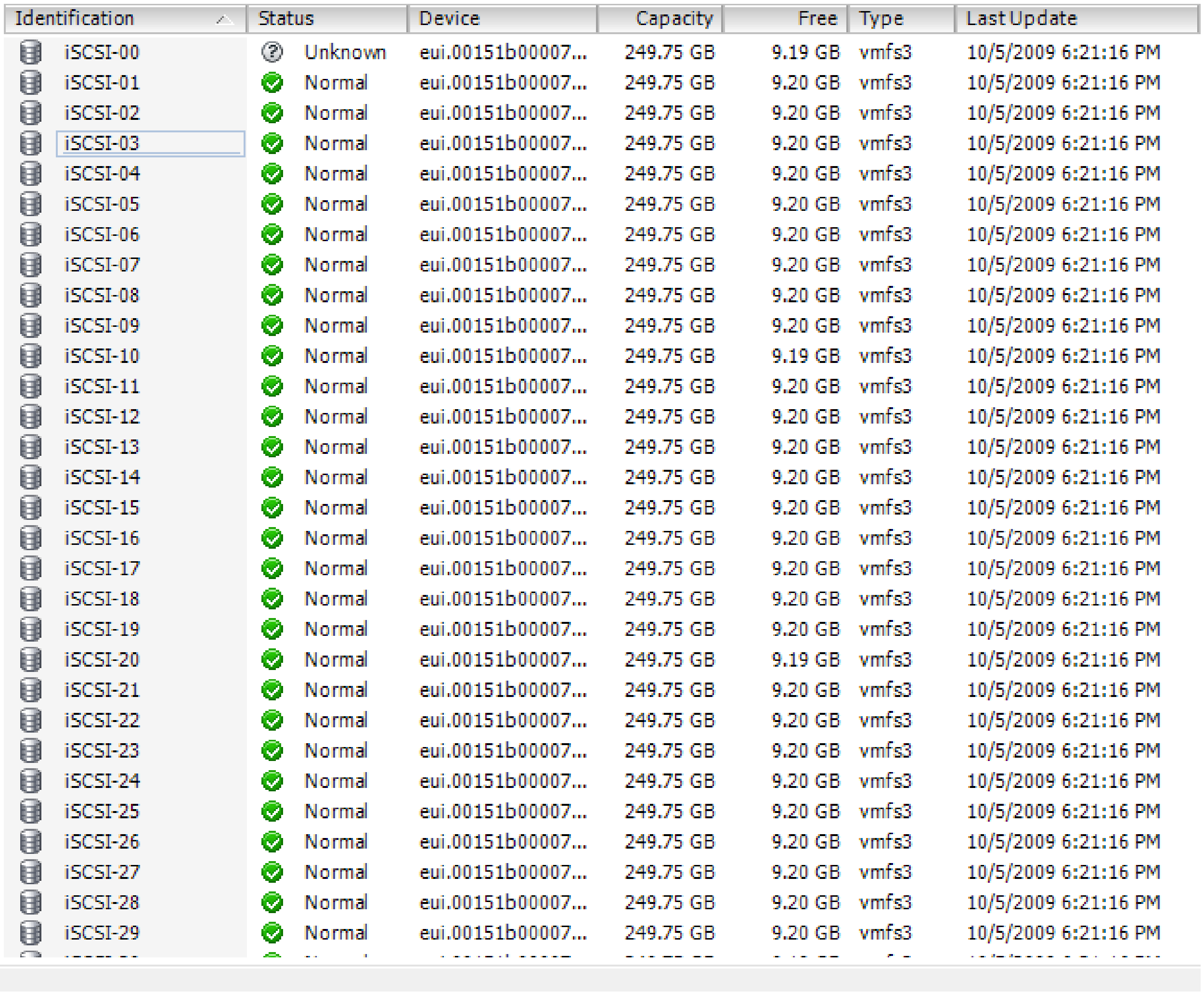
Adding iSCSI Datastores

To create a VMFS datastore on an iSCSI LUN use the **Add Storage** wizard

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Hardware**, select **Storage** (SCSI, SAN and NFS).
2. Click **Add Storage**.
3. Select the **Disk/LUN** storage type and click **Next**.
4. The **Select Disk/LUN** page appears. This can take a few seconds depending on the number of targets.
5. Select the iSCSI device to use for your datastore and click **Next**.
6. Review the current disk layout and click **Next**.
7. Enter a datastore name and click **Next**. The datastore name appears in the vSphere Client, and the label must be unique within the current VMware vSphere instance.
8. If needed, adjust the file system values and capacity you use for the datastore. By default, the entire free space available on the storage device is offered to you.
9. Click **Next**. The **Ready to Complete** page appears.
10. Review the datastore configuration information and click **Finish**.

A datastore is now available on the iSCSI storage device.

VMFS datastores will be presented in the vSphere client by selecting the **Storage** link in the **Hardware** settings of the desired ESX host. Any other ESX host with a fully configured iSCSI software initiator that has access to the same set of iSCSI targets will see the same list of VMFS datastores and raw LUNs upon a rescan. VMFS datastores are shared across all ESX hosts for both vMotion, DRS and HA purposes.



Identification	Status	Device	Capacity	Free	Type	Last Update
iSCSI-00	Unknown	eui.00151b00007...	249.75 GB	9.19 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-01	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-02	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-03	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-04	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-05	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-06	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-07	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-08	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-09	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-10	Normal	eui.00151b00007...	249.75 GB	9.19 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-11	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-12	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-13	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-14	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-15	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-16	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-17	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-18	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-19	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-20	Normal	eui.00151b00007...	249.75 GB	9.19 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-21	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-22	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-23	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-24	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-25	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-26	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-27	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-28	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM
iSCSI-29	Normal	eui.00151b00007...	249.75 GB	9.20 GB	vmfs3	10/5/2009 6:21:16 PM

Figure 32: ESX list of VMFS datastores after iSCSI initiator rescan

iSCSI Initiator Multi-Pathing Configuration

Both iSCSI VMFS datastores and RDMs can use multiple paths to provide redundancy across all network components in the iSCSI data path from ESX hosts network interfaces, to switches, and Isilon cluster nodes, and interfaces. An Isilon cluster in turn provides a multi-node **active/active** storage cluster with multiple storage targets that can be used for both redundancy and aggregate I/O using the **FIXED** path selection policy. General networking best practices is provided in the section **OneFS Networking Best Practices for iSCSI** in the previous chapter on **Isilon iSCSI Configuration.** This section discusses multi-pathing configuration guidelines specific to ESX iSCSI software initiator.

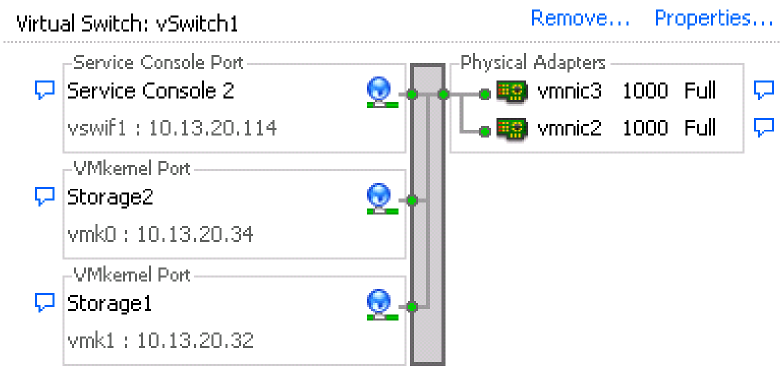
Note: iSCSI Multi-pathing is not currently supported with Distributed Virtual Switches on the VMware offering or the Cisco Nexus 1000V. VMware and VMware are working to fix this and allow any virtual switch to be supported.

Because iSCSI multi-pathing does not have direct access to physical network interfaces, you first need to connect each physical interface to a separate VMkernel port, and then associate all VMkernel ports with the software iSCSI initiator using a **port binding** technique. As a result, each VMkernel port connected to a separate network interface becomes a different path that the iSCSI software initiator can use.

Note: Hardware iSCSI adaptors have not been tested and are not currently supported for use with Isilon storage.

The following procedure uses two or more network interfaces assigned to iSCSI storage traffic within a single vSwitch. In this configuration you associate each VMkernel port with a network single network interfaces:

1. In the vSphere Client, select the ESX host **Configuration** tab and under **Hardware**, select **Networking**.
2. Select an existing **vSwitch** for iSCSI storage or create a new one using the **Add Networking** wizard to create a new vSwitch.
    a. Connect additional network interfaces to the vSwitch.
    b. In the vSwitch **Properties** dialog box, click the **Network Adapters** tab and click **Add**.
    c. Select one or more adapters from the list and click **Next**.
    d. Review the information on the Adapter Summary page, and click Finish.
3. Create VMkernel ports for all network adapters that you connected. The number of VMkernel ports must correspond to the number of network adapters on the vSwitch.
    a. In the vSwitch **Properties** dialog box, click the **Ports** tab and click **Add**.
    b. Select **VMkernel** and click **Next**.
    c. Under **Port Group Properties**, enter a **network label** and click **Next**.
    d. Specify the **IP settings** and click **Next**. When you enter subnet mask, make sure that the network adapter is set to the subnet of the storage system it connects to.
    e. Review the information and click **Finish**.
4. By default, for each VMkernel port on the vSwitch, all network interfaces appear as active. You must override this setup, so that each port maps to only one corresponding active adapter.
    a. On the **Ports** tab, select a VMkernel port and click **Edit**.
    b. Click the **NIC Teaming** tab and select **Override vSwitch failover order**. Designate only one adapter as active and move all remaining adapters to the **Unused Adapters** category.
5. Repeat the last step for each VMkernel port on the vSwitch.



Virtual Switch: vSwitch1

Service Console Port
Service Console 2
vswif1 : 10.13.20.114

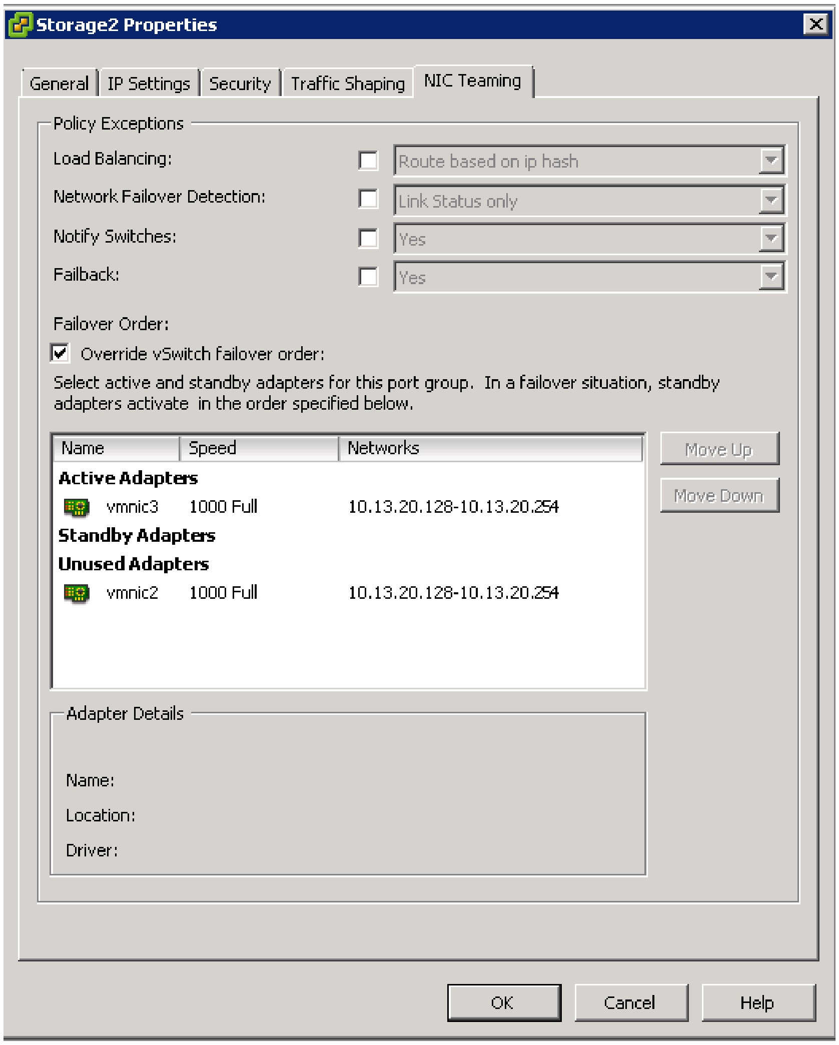
VMkernel Port
Storage2
vmk0 : 10.13.20.34

VMkernel Port
Storage1
vmk1 : 10.13.20.32

Physical Adapters

vmnic3	1000	Full
vmnic2	1000	Full

Figure 33: Two VMkernel ports in a vSwitch



Storage2 Properties

Policy Exceptions

Load Balancing: □ Route based on ip hash

Network Failover Detection: □ Link Status only

Notify Switches: □ Yes

Failback: □ Yes

Failover Order:

☑ Override vSwitch failover order:

Select active and standby adapters for this port group. In a failover situation, standby adapters activate in the order specified below.

Name	Speed	Networks
Active Adapters		
vmnic3	1000 Full	10.13.20.128-10.13.20.254
Standby Adapters		
Unused Adapters		
vmnic2	1000 Full	10.13.20.128-10.13.20.254

Adapter Details

Name:

Location:

Driver:

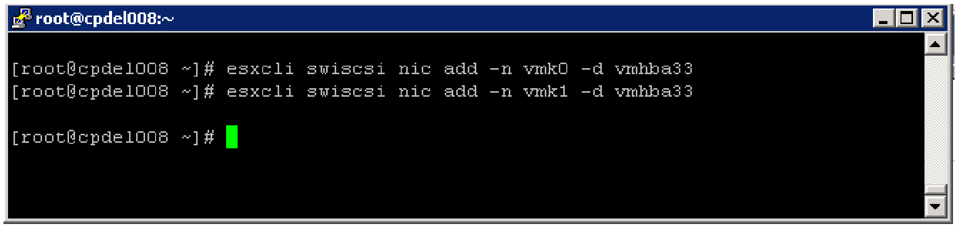
Figure 34: Only one active adapter assigned to VMkernel port

Binding iSCSI Software Initiator to VMkernel Ports

Once VMkernel ports are associated with their respective network interface the final step for activating iSCSI multi-pathing involves binding the iSCSI software initiator to those VMkernel ports:

1. Identify the VMkernel ports names assigned to physical adapters those are located below the port label next to the VMkernel assigned IP address (as shown above in figure 24 as vmk0 and vmk1).
2. Connect to the ESX service console and execute the following commands (one for each VMkernel) **`esxcli swiscsi nic add -n <port_name> -d <vmhba>`**
3. Verify the binding status by executing the following command **`esxcli swiscsi nic list -d <vmhba>`**

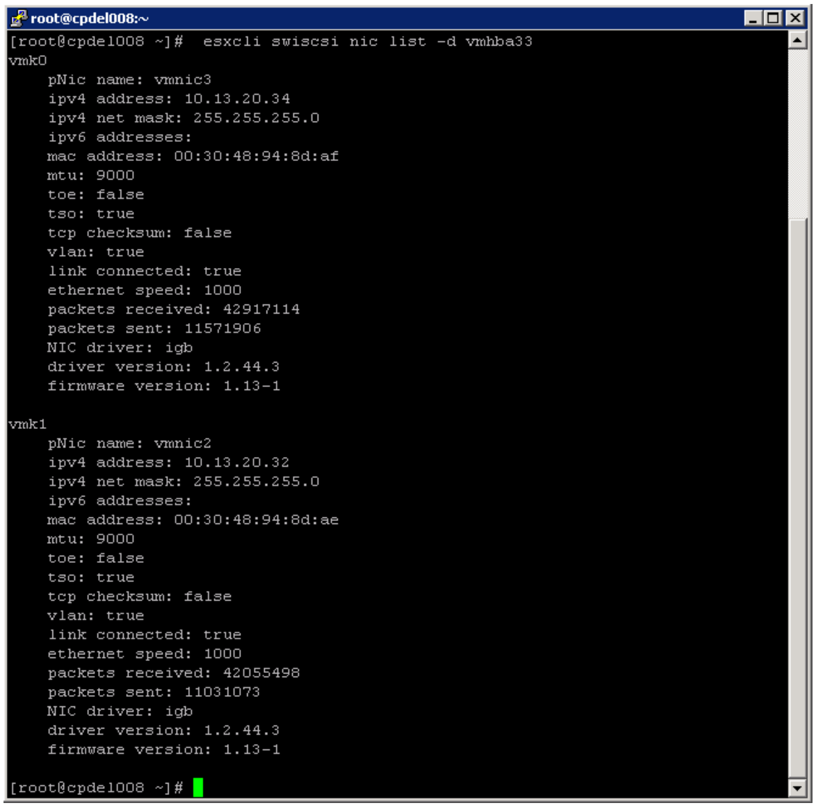
See next set of figures for an example.



```
[root@cpdel008 ~]# esxcli swiscsi nic add -n vmk0 -d vmhba33
[root@cpdel008 ~]# esxcli swiscsi nic add -n vmk1 -d vmhba33

[root@cpdel008 ~]#
```

Figure 35: Binding iSCSI software initiator to VMkernel ports



```
[root@cpdel008 ~]#  esxcli swiscsi nic list -d vmhba33
vmk0
    pNic name: vmnic3
    ipv4 address: 10.13.20.34
    ipv4 net mask: 255.255.255.0
    ipv6 addresses:
    mac address: 00:30:48:94:8d:af
    mtu: 9000
    toe: false
    tso: true
    tcp checksum: false
    vlan: true
    link connected: true
    ethernet speed: 1000
    packets received: 42917114
    packets sent: 11571906
    NIC driver: igb
    driver version: 1.2.44.3
    firmware version: 1.13-1

vmk1
    pNic name: vmnic2
    ipv4 address: 10.13.20.32
    ipv4 net mask: 255.255.255.0
    ipv6 addresses:
    mac address: 00:30:48:94:8d:ae
    mtu: 9000
    toe: false
    tso: true
    tcp checksum: false
    vlan: true
    link connected: true
    ethernet speed: 1000
    packets received: 42055498
    packets sent: 11031073
    NIC driver: igb
    driver version: 1.2.44.3
    firmware version: 1.13-1

[root@cpdel008 ~]#
```

Figure 36: Verifying binding of iSCSI software initiator to VMkernel ports

Path Management for Load Balancing

Balancing load among available paths increases availability and can also improve performance by distributing load on one or more LUNs accessed through multiple paths in parallel. With **active/active** storage systems, you can configure your ESX hosts to load balance traffic across multiple adapters by assigning preferred paths to your LUNs. Path selection policy must be set to “**FIXED**” to be able to specify a preferred path and fail back to it when it becomes available again after a failover event. An alternative method is to use an **active/passive** storage system with path selection set to “**MRU**” (most recently used) but this type of path selection will not fail back to the preferred path and require administrative intervention to rebalance load across all available paths.

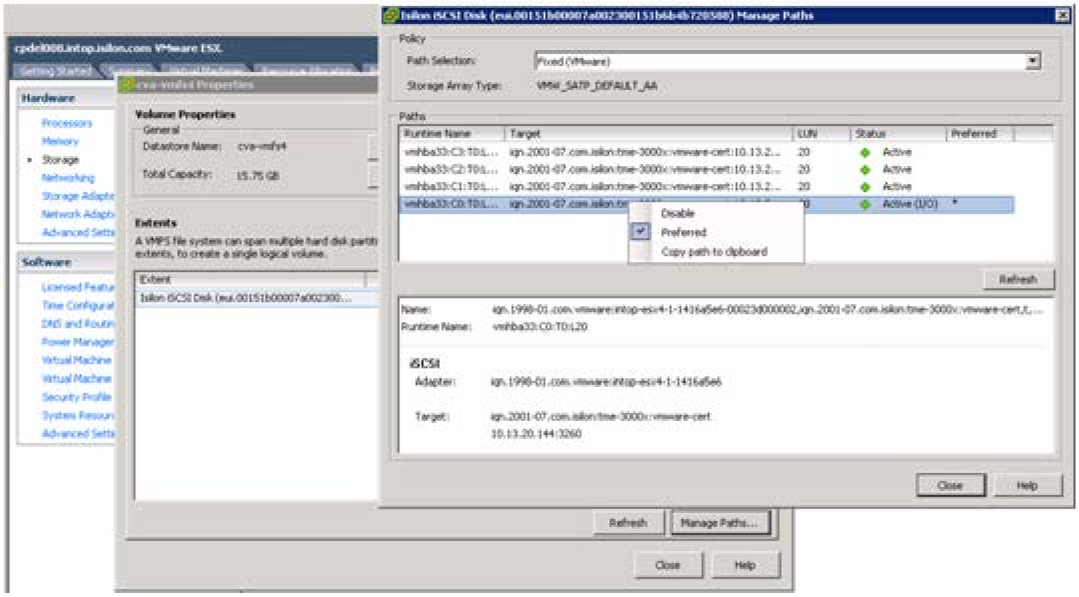
Isilon further enhances load balancing by providing on-the-fly scalability of nodes that each act as an iSCSI storage processor/target so as you add more nodes to the cluster you increase both capacity and performance.

Note: While Isilon supports the **active/active** storage plug-in type with **fixed** path selection policy, Isilon recommends that all ESX hosts accessing the same datastore use the same preferred path to eliminate any cross-node lock contention. Use of round robin (RR) path selection policy is considered experimental at this time.

Change the Path Selection Policy

Generally, you do not have to change the default multi-pathing settings your ESX host uses. However, if you want to make any changes, you can use the **Manage Paths** dialog box to modify a path selection policy and specify the preferred path for the **Fixed** policy.

1. In the vSphere Client, select the ESX server machine, then select the **Configuration** tab. Under **Hardware**, select **Storage** (SCSI, SAN and NFS).
2. Right click on a **Datastore** or **LUN device** and click on **Properties**.
3. Open the **Manage Paths** dialog box either from the datastores **Properties** dialog box or **Devices** view.
4. Select a path selection policy from the drop down selection box.
5. For the fixed policy, specify the preferred path by right-clicking the path you want to assign as the preferred path, and selecting **Preferred (also marked with *)**.



Isilon iSCSI Disk (eui.00151b00007a0023001 51b6b4b720588) Manage Paths

Policy

Path Selection: Fixed (VMware)

Storage Array Type: VMW_SATP_DEFAULT_AA

Paths

Runtime Name	Target	LUN	Status	Preferred
vmhba33:C3:T0:L...	iqn.2001-07.com.isilon:tme-3000x:vmware-cert:10.13.2...	20	Active	
vmhba33:C2:T0:L...	iqn.2001-07.com.isilon:tme-3000x:vmware-cert:10.13.2...	20	Active	
vmhba33:C1:T0:L...	iqn.2001-07.com.isilon:tme-3000x:vmware-cert:10.13.2...	20	Active	
vmhba33:C0:T0:L...	iqn.2001-07.com.isilon:tm...	20	Active (I/O)	*

Name: iqn.1998-01.com.vmware:intop-esx4-1-1416a5e6-00023d000002,iqn.2001-07.com.isilon:tme-3000x:vmware-cert,t,...

Runtime Name: vmhba33:C0:T0:L20

iSCSI

Adapter: iqn.1998-01.com.vmware:intop-esx4-1-1416a5e6

Target: iqn.2001-07.com.isilon:tme-3000x:vmware-cert
10.13.20.144:3260

cva-vmfs4 Properties

Volume Properties

General

Datastore Name: cva-vmfs4

Total Capacity: 15.75 GB

Extents

A VMFS file system can span multiple hard disk partit... extents, to create a single logical volume.

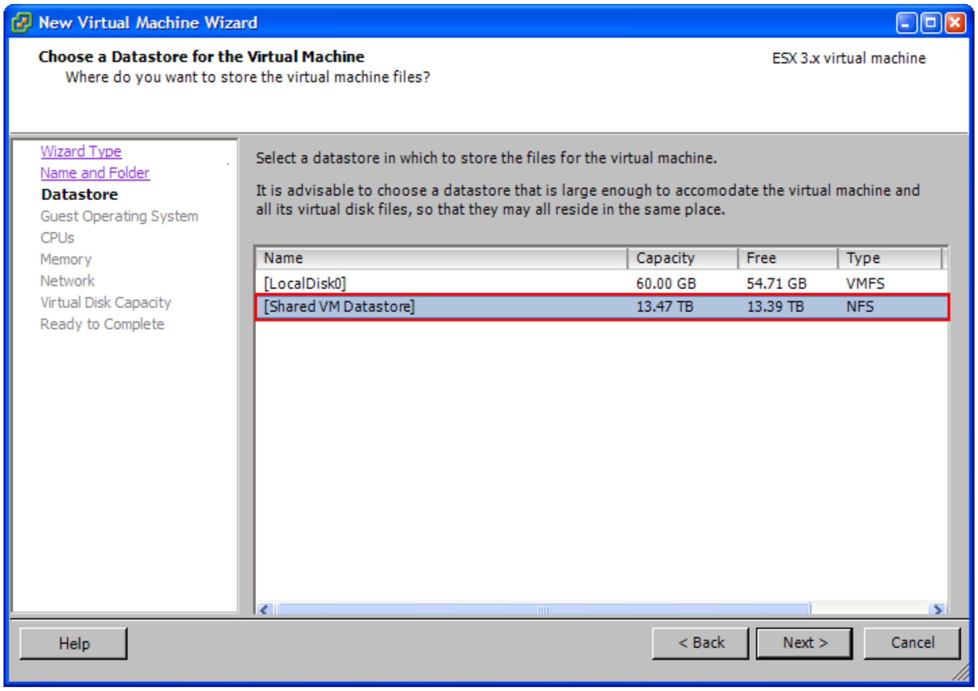
Extent
Isilon iSCSI Disk (eui.00151b00007a002300...

Figure 37: Setting datastore or LUN path selection policy

7. Virtual Machine Configuration

VMDKs and Other VM Files on Isilon Clusters

Once NFS or iSCSI datastores are created on an Isilon IQ storage cluster, virtual machines can be created or migrated to or from other storage systems. When creating a virtual machine, select a datastore on the Isilon cluster in the **New Virtual Machine** wizard, as in the figure below.



Choose a Datastore for the Virtual Machine

Where do you want to store the virtual machine files?

Select a datastore in which to store the files for the virtual machine.

It is advisable to choose a datastore that is large enough to accomodate the virtual machine and all its virtual disk files, so that they may all reside in the same place.

Name	Capacity	Free	Type
[LocalDisk0]	60.00 GB	54.71 GB	VMFS
[Shared VM Datastore]	13.47 TB	13.39 TB	NFS

Figure 38: Selecting a virtual machines datastore on an Isilon cluster

When using NFS datastores on an Isilon cluster, virtual machines are managed directly by OneFS and can be located by navigating to the virtual machine directory path on OneFS. Each of the datastore subdirectories represents a single virtual machine instance along with .vmx config file, .vmdk virtual disk files and other files.

When using an iSCSI datastore on an Isilon cluster, virtual machines cannot be directly located on OneFS and are only viewable through the vSphere user interfaces or ESX service console. The LUNs that make up the datastore can be located in the OneFS file system but they provide little information on the specific virtual machines stored in them.

Creating Virtual Disks in a Datastore on Isilon IQ Cluster

When creating a new virtual machine the first virtual hard disk is created along with the virtual machine and stored in the same datastore as the virtual machine is stored in. The datastore can be either an NFS datastore or an iSCSI datastore. Additional virtual hard disks can be added to existing virtual machines and the corresponding VMDK files can be stored in the same datastore as the virtual machine or other datastores accessible by the ESX host.

1. In the vSphere client, select the virtual machine to which you want to add disk, then select **Edit Settings**.
2. On the **Hardware** tab of the Virtual Machine **Properties** dialog, click **Add**.
3. In the **Add Hardware** Wizard, select **Hard Disk** as the device type, and click **Next**.
4. On the **Select a Disk** screen, select “**Create a new virtual disk**”. Click **Next**.
5. On the **Disk Capacity** screen, enter a size for the disk and select either to store the disk files with the virtual machine, or to locate them on a separate datastore. In either case, you can place the virtual disk on a datastore residing on the cluster.

Using iSCSI Datastores and iSCSI Raw LUNs for Raw Device Mapping

Raw device mapping (RDM) allows a special file in a VMFS volume to act as a proxy for a raw LUN. The mapping file contains metadata used to manage and redirect disk accesses to the physical device. The mapping file which can be thought of as a symbolic link from a VMFS datastore to a raw LUN, in effect, merges VMFS manageability with raw LUN access. RDMs are useful when you want to dedicate a raw LUN to a single virtual machine without sharing it with other virtual machines in the datastores. RDMs are also useful for native clustering technologies such as Microsoft clustering services (MSCS) or clustering between physical and virtual machines.

RDM disk are creating by selecting the **Edit Settings** option of a virtual machine and selecting the **Raw Device Mapping** in the **Add→Hard Disk** wizard. When adding an RDM you must choose a datastore to keep the mapping file in and a compatibility mode:

- **Physical compatibility mode -** the virtual machine can access the LUN directly. This is generally used from the application inside VM wants to directly access LUN. However using physical compatibility mode you lose the ability to clone the virtual machine, creating virtual machines from a template, or storage migration of virtual disks.
- **Virtual compatibility mode** – allows the LUN to behave as VMDK, which enables to use features like cloning to template, cloning to VM or storage migrations of virtual disks.

Note: When you wish to implement **Microsoft clustering** you have to select **Physical Compatibility** mode.

Setting Guest Operating System Timeout for iSCSI Failover

iSCSI path failover occurs when the active path to a LUN is changed from one path to another, usually because of some network component failure along the current path. During the failover process I/O might pause for 30 to 60 seconds until the ESX iSCSI initiator determines that the link is unavailable and until failover is complete. As a result, the virtual machines (with their virtual disks installed on the Isilon cluster) can appear unresponsive. If you attempt to display the ESX host, its storage devices, or its adapter, the operation might appear to stall. After failover is complete, I/O resumes normally. If none of the connections to the storage device is working, some virtual machines might encounter I/O errors on their virtual SCSI disks. For Windows 2000 and Windows Server 2003 guest operating systems, you can increase operating system timeout by editing the registry.

1. Back up your Windows registry.
2. Select **Start** > **Run**.
3. From the command prompt type **regedit.exe**, and click **OK**
4. In the left-panel hierarchy view, double-click **HKEY_LOCAL_MACHINE→System→CurrentControlSet→Services→Disk**.
5. Select the **TimeOutValue** and set the data value to x03c (hexadecimal) or 60 (decimal).
6. Click **OK** to exit the **Registry Editor**.

Virtual Machine Disk Alignment

Virtual machine hard disks are stored as VMDK files on NFS or iSCSI VMFS datastores. Virtual machines use their virtual hard disks to store data by formatting them into local partitions and file systems. If guest OS partitions are not to properly align with the shared storage block boundaries performance may suffer. Isilon OneFS aligns data at 8K block boundaries and if VMDK partitions are not aligned at 8K boundaries additional I/O operations are required to serve VMDK access requests. Isilon and VMware recommend the following to avoid issues with VMDK alignment:

- VMDK misalignment related performance issues are mostly visible in random I/O operations so Isilon recommends creating separate VMDKs for OS boot partitions, (which are mostly used for reading OS and applications binaries), and data partitions (which are used for random I/O applications and user data access). In this manner VMDKs for boot partitions, which are allocated when creating a new virtual machine, do not need to be further aligned.
- If boot partitions are to be aligned it should be done before assigning them to a new virtual machine by using another virtual machine as a surrogate for the new VMDK. This is accomplished by adding a new virtual disk to the surrogate virtual machine and aligning the VMDK using the methods below. Next disassociate the VMDK from the surrogate virtual machine and add it to the new virtual machine by choosing an existing VMDK for the new virtual machine hard disk.
- Another way to address boot partition VMDK alignment is by creating a template virtual machine with an aligned boot VMDK and cloning it to create new virtual machines.

VMDK alignment can be done either by using virtual machine native OS tools (after boot partition already exists) or by using the ‘fdisk’ tool in the ESX service console.

Creating Well Aligned Disk Partitions in a Windows Virtual Machine

Once you have created a new Windows virtual machine with a hard disk or added hard disks to an existing Windows virtual machine, you should be able to see each those disks as devices in the Windows **Disk Management** tool. However, if you create partitions and format the partitions in disk management you will **NOT** achieve the best performance possible. There are two reasons for this:

1. Disk alignment and partition. In versions of Windows® prior to Windows 2008, when creating the first partition of a disk, the first sector of that partition is not, by default, aligned with the physical disk. There are also times when Windows 2008 does not align the disk properly. Therefore, it is highly recommended that when creating partitions you use '**diskpart'** utility with the **‘align=64’** option.
2. Block size. Isilon uses an 8KB block size. Windows typically formats in a 4KB block size. To get the best performance you will want your disks also formatted with an 8K block size. This is also called the unit allocation size. When formatting the partition, we recommend using the command line interface to do so.

To Create and Format a Disk:

1. Open a command prompt, type DISKPART, and press **ENTER.**
2. **For Windows 2008**, Type the following commands within **DISKPART**:

```
SELECT DISK 1
CREATE PARTITION PRIMARY ALIGN=64
SELECT PARTITION 1
FORMAT FS=NTFS LABEL="New Volume" QUICK UNIT=8192
ASSIGN LETTER=E
EXIT
```

3. **For Windows 2003**, the format command is not available within **DISKPART**. Type the following commands in DISKPART:

```
SELECT DISK 1
CREATE PARTITION PRIMARY ALIGN=64
SELECT PARTITION 1
ASSIGN LETTER=E
EXIT
```

Then type this command from the command prompt:

```
FORMAT E: /FS:NTFS /V:NewVolume /Q /A:8192
```

Checking Disk Alignment and Block Size in a Windows Virtual Machine

► To Check an Existing Partition for Alignment:

1. Open a command prompt and type:

   ```
   wmic partition get BlockSize, StartingOffset, Name, Index
   ```

2. Notice the result of StartingOffset is 65536 (64KB):

   ```
   BlockSize  Index  Name                   StartingOffset
   512        0      Disk #1, Partition #0  65536
   ```

Creating Well Aligned Partitions Prior to Installing a Windows Virtual Machine

Virtual disks can be formatted with the correct offset at the time of creation by simply booting the VM before installing an operating system and manually setting the partition offset. For Windows guest operating systems, consider using the Windows Preinstall Environment boot CD or alternative "live dvd" tools. To set up the starting offset, follow the instructions above.

Aligning VMDK from the ESX Service Console

VMDKs can also be aligned by using the ‘fdisk’ utility in the ESX service console. This procedure can be used for VMDKs stored both in iSCSI VMFS and NFS datastores regardless of the guest OS to be installed on them:

1. Log in to the ESX service console.
2. Change directory to the virtual machine under the ESX VMFS volume directory:

   ```
   cd /vmfs/volumes/<datastore>/<VM home dir>
   ```

   ```
   ls –l
   ```

3. Indentify the number of cylinders in the virtual disk by reading the VMDK descriptor file. Look for the line **ddb.geometery.cylinders** by typing:

   ```
   grep cylinders <virtual machine>.vmdk
   ```

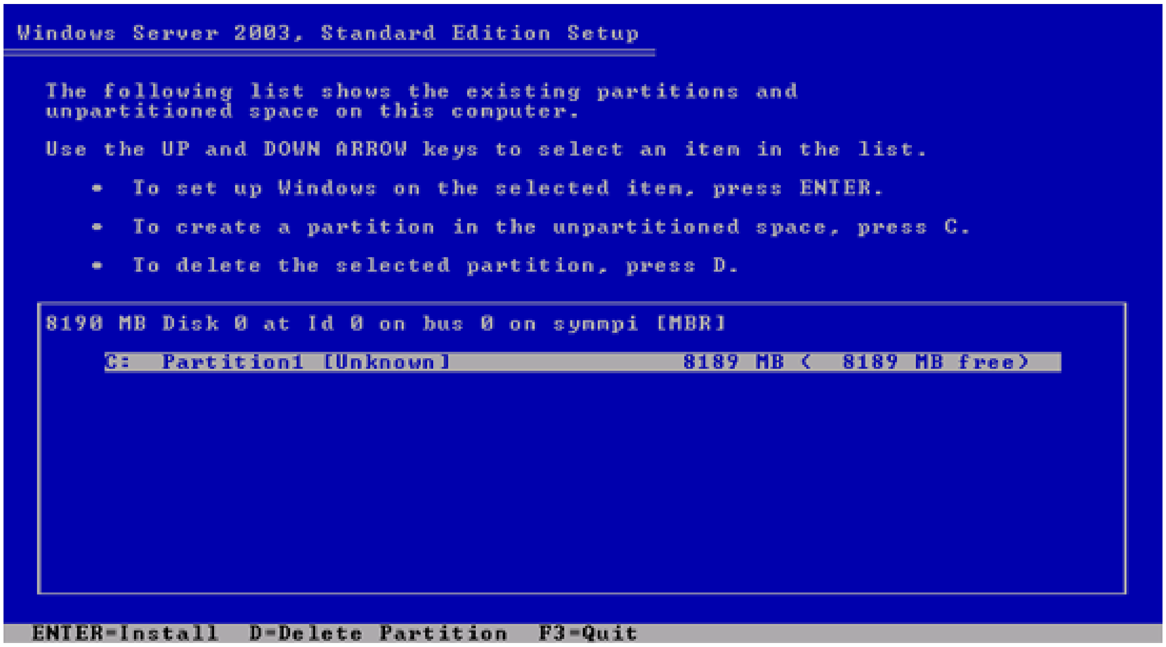
4. Run **fdisk** on the VMDK flat file (the –flat.vmdk file) by typing:

   ```
   fdisk <virtual machine>-flat.vmdk
   ```

5. Once in fdisk, enter **Expert Mode** by typing **x** and pressing **Enter**.
6. Type **c** to enter the number of cylinders and press **Enter**.
7. Enter the number of cylinders that you found from step 3.
8. Type **p** at the expert command screen to look at the **partition table**. The results should be a table of all zeros.
9. Return to **Regular Mode** by typing **r**.
10. Create a new partition by typing **n** and then **p** for the partition type.
11. Enter **1** for the **partition number**, **1** for the **first cylinder**, and press **Enter** for the **last cylinder** to use the default value.
12. Type **x** for **Extended Mode** to set the starting offset.
13. Type **b** for **starting offset** and press **Enter**

14. Selecting **1** for the partition and press **Enter**. Enter **64** and press **Enter**. The value 64 represents the number of 512 byte sectors used to create a starting offset of 32,768 bytes (32KB).
15. Check the partition table by typing **p**. If you did this correctly the top row of the output should display disk geometry including the starting offset of 64. The value of 64 is just an example (and the most common value) but any value that is divisible by 16 is valid (represents 16 512 byte sectors that align with 8KB blocks).
16. Return to **Regular Mode** by typing **r**.
17. Type **t** to set the **system type** to **HPFS/NTFS** for Windows partitions.
18. Enter **7** for the hexcode for HPFS/NTFS. Type **L** for a list of other OS partition types such as **Linux Extended** or **Linux LVM**.
19. Save and write the partition by typing w. Ignore the warning, as this is normal.
20. Start the virtual machine and run Windows (or other OS installation). During the installation process you will be prompted that a partition exists. Select the existing partition to install the OS into.

Note: do not destroy or recreate the partition or you will lose the sector alignment



```
Windows Server 2003, Standard Edition Setup

   The following list shows the existing partitions and
   unpartitioned space on this computer.

   Use the UP and DOWN ARROW keys to select an item in the list.

       •  To set up Windows on the selected item, press ENTER.

       •  To create a partition in the unpartitioned space, press C.

       •  To delete the selected partition, press D.

   8190 MB Disk 0 at Id 0 on bus 0 on symmpi [MBR]

        C:   Partition1 [Unknown]                8189 MB (   8189 MB free)

  ENTER=Install   D=Delete Partition   F3=Quit
```

Figure 39: Selecting an existing partition during Windows installation

Note: further information on iSCSI VMFS partition alignment for VI 3 is available in: http://www.vmware.com/pdf/esx3_partition_align.pdf

8. Migrating Virtual Machines between ESX Hosts

Once the Isilon cluster is configured and available to ESX server machines using iSCSI VMFS and NFS datastores, virtual machines located on those hosts can be migrated without further configuration.

With vSphere 4.1 VMware provides the option of either migrating the virtual machine to a different ESX host (vMotion), a different datastore (Storage vMotion), or both a different ESX host and a different datastore.

Cold Migrations

A cold migration involves moving a virtual machine between hosts while it is powered off. This type of migration consists primarily of ‘relocating’ VM files located on shared storage and changes to the VM config files to reflect the new host where it resides.

Migrating with vMotion

vMotion allows a running VM to be migrated between ESX hosts. vMotion requires the hardware of the two ESX servers to be compatible and the two hosts must be able to access shared storage, as with cold migrations.

When using vMotion with Isilon storage, the two ESX hosts participating in the migration must use the same datastore to access the VM:

- For NFS datastores the node IP address and directory of the datastore must be the same, e.g. if host 1 points to 192.168.0.1/ifs/vmware, then host 2 must also point to the same path.
- For iSCSI VMFS datastore the second ESX host iSCSI initiator must have access to the same iSCSI LUN (CHAP and target access restrictions) and must have the VMFS datastores mounted after a rescan.

Following best practices for network design (as detailed above) will ensure both ESX hosts see the same datastore.

If NFS failover is being used, and the hosts involved in the migration are using the dynamic IP address for the shared datastore, then a migration won’t be interrupted or impacted by a failure on the node to which the hosts are connected.

Migration with Storage vMotion (sVMotion)

vSphere 4.1 has the ability to migrate all the contents of a virtual machine (VMDKs and configuration files) to another storage location while the virtual machine is still running using **sVMotion**. This simplifies the process of migrating from one storage infrastructure to another. sVMotion also facilitates migration between various types of datastores (SAN, iSCSI and NFS) across different types of storage systems, as well as changing the configuration of the VMDKs (thin to thick provisioning).

If a virtual machine needs to be migrated within the same Isilon cluster between two different NFS datastores, doing it while it is shutdown is a faster and less impactful way:

1. Ensure the VM is not running and unregister from the datastore by right-clicking and selecting **un-register**.
2. Move the VM directory to a new location on the cluster where the target datastore has access to.
3. Register the VM in the new datastore. By browsing the datastore selection the .vmx file of the VM Add it to Inventory.

9. Working with Snapshots

Snapshots Explained

A snapshot of a virtual machine is simply a point-in-time copy of the state of the virtual machine, including memory, disk contents and virtual machine settings. Because a virtual machine, stored in an NFS datastore, consists of a set of files located on an Isilon cluster, it's possible to take snapshots using the VMware snapshot utility or Isilon's SnapshotIQ. Virtual machines hosted in an iSCSI datastore can either use raw device mapping LUNs to take snapshots of raw LUNs on an Isilon cluster or use the VMware snapshot utility.

Snapshot Data Consistency Considerations

When taking a snapshot of a virtual machine, it's critical to consider the state of the VM and the application activity. Taking a snapshot of a running VM, especially one that is running applications that depend on data consistency, (such as a database), can result in data corruption or missing transactions. For example, if a VM is reading or writing a large file and a snapshot is taken before this I/O is complete, (if the VM is reverted to the snapshot), the file that was being read or written will be corrupt and will have to be recovered using another mechanism, such as a tape backup. This data consistency issue is present regardless of what snapshot technology used, whether it is VMware snapshot, Isilon SnapshotIQ or some other storage or host based snapshot technology. It is also highly dependent on the OS type and version. Depending on the quiescing, mechanism the snapshot can be:

- **Crash-consistent**: this snapshot relies on the operation systems ability to survive sudden crashes or reboots and allows the operating system to restart without corruption. Most modern operating systems such as Windows and Linux are crash-consistent.
- **File-system consistent:** a snapshot that integrates with file system capabilities to flush its cache before a snapshot is taken. VMware provides a sync driver installed in every guest operating system with the VMware tools package. To take advantage of the VMware sync driver a virtual machine should be quiesced before a OneFS snapshot is taken. This can be accomplished using VMware scripts that does the following:
    - First take a VMware snapshot to quiesce the virtual machine.
    - Next, take an OneFS snapshot for a point in time copy of the virtual machine directory or iSCSI LUN it resides in
    - Finally, the VMware snapshot can be removed so that all that remains is a OneFS snapshot.
- **Application consistent**: similar to file system consistent snapshots, only in this case, the OneFS snapshot needs some mechanism to communicate with the applications running inside the virtual machines to complete any outstanding application I/O and clear application specific cache. A file-system consistent snapshot is not enough in this case because applications often have their own data consistency logic or do not use a file system at all and access block devices directly (in this case the virtual hard disk or RDM). Starting with ESX 3.5 U2 and above VMware integrates with Window Virtual Shadow Service (VSS) to allow application providers to supply VSS writers that are triggered by the VSS quiescing mechanism. In this way an ESX server taking a VMware snapshot can trigger the VSS mechanism in a Windows VM and provide application level consistency. In this manner OneFS snapshots can be integrated in the same way as described for file-system consistency snapshots.

  Note: care must be taken to ensure that VSS applications writers are properly installed inside the virtual machine in addition to the generic VMware supplied VMware tools.

A good resource for fully understanding VMware snapshot capabilities can be found in VMware's "Virtual Machine Backup Guide" which can be found here: http://www.vmware.com/pdf/vsphere4/r40/vsp_vcb_15_u1_admin_guide.pdf

Taking Snapshots

Offline Snapshots (Cold)

An offline snapshot is taken with the VM guest OS shut down and the VM powered off. Taking snapshots in this manner ensures data consistency for applications inside the VM. The tradeoff is that for production virtual machines, it may not be possible to completely shut down operations in order to take a snapshot. Offline snapshots are nearly instantaneous, and a VM is typically much faster to boot than an equivalent physical server, so downtime is minimized.

Nearline Snapshots (Warm)

A nearline snapshot is taken while the VM is in the suspended state. Nearline snapshots, like offline snapshots, generally take less than a second to complete, and when the VM is returned to a powered on state, processing resumes from the point in time at which the VM was suspended. Nearline snapshots can have the same potential issues with data consistency as online snapshots.

Online Snapshots (Hot)

An online snapshot is taken while the VM is running, and possibly has applications running inside it. Online snapshots allow the VM and its applications to continue running, so there is no interruption of service; however, as noted above, if applications are reading or writing data, there is no guarantee of data consistency if the VM is reverted to an online snapshot.

Taking Snapshots with the ESX Snapshot Utility

The procedure for taking snapshots with the ESX snapshot utility is the same for all three types of snapshot.

1. Confirm that the VM you wish to snapshot is in the desired state (running, suspended or powered off).
2. In the vSphere client, select the VM in the left-side tree view, then select the **Take Snapshot** command by either clicking the toolbar button or by right-clicking on the VM and selecting **Snapshot→Take Snapshot** from the context menu.
3. In the **Take Snapshot** dialog, provide a name for the snapshot and an optional description. If taking an online snapshot, select whether to include the VM’s memory contents in the snapshot. Click **OK** to take the snapshot.

Reverting To a Snapshot

To revert to the last snapshot taken:

- In the vSphere client, select the VM in the left-side tree view, then select the **Revert to Snapshot** command by either clicking the toolbar button or by right-clicking on the VM and selecting **Snapshot→Revert to Snapshot** from the context menu.

To revert to a specific snapshot:

- In the vSphere client, select the VM in the left-side tree view, then open the **Snapshot Manager** by either clicking the toolbar button or by right-clicking on the VM and selecting **Snapshot→Snapshot Manager** from the context menu.

In the Snapshot Manager, select the snapshot you want to revert to, then click OK. For more details on reverting to a specific snapshot, please see the VMware ESX Server 3 Administration Guide.

10. Isilon for vCenter 1.0 Plugin

Integrating Isilon SnapshotIQ and VMware Snapshots via Isilon for vCenter 1.0

It is often beneficial to integrate Isilon SnapshotIQ snapshots with VMware snapshots for the following reasons:

1. VM Snapshots + Isilon SnapshotIQ snapshots provide more flexibility in data management for VMware environments.
2. SnapshotIQ snapshots can be tied into a storage backup application which enables you to capture the VMware snapshot to tape. They are often tied in as a pre-exec process to the backup application and gives you the ability to mirror from the storage snapshot to a different location.
3. SnapshotIQ snapshots allow you to keep a larger number of snapshots vs. having a large number of VM snapshots, which can get unwieldy and are not purged on a regular basis.
4. SnapshotIQ snapshots can be taken automatically at any interval, and retention policies can be applied to delete the snapshots based on expiration dates/times. There is no way to setup a snapshot schedule in VMware or setup the configuration to automatically delete a snapshot after a certain amount of time (this could be done via the VMware vStorage API with custom coding).

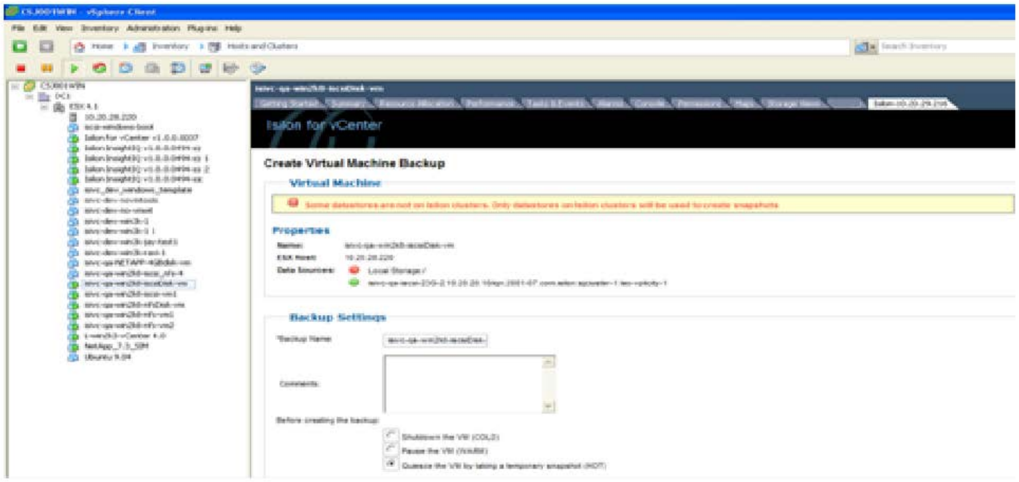
Isilon for vCenter 1.0 Plugin

Isilon for vCenter is comprised of two components:

- Virtual appliance running an Isilon virtualization server
- VMware vCenter WebUI plugin module.

The virtual appliance uses VMware APIs to interface to VMware vCenter and the Isilon Platform API to interface to the cluster in order to orchestrate backups and restores across the Isilon storage environments. Finally, the Isilon plugin hooks into vCenter and presents a management screen to control VM backup and restore functionality.

Isilon for vCenter is accessed via a tab in the vCenter console, or by right clicking on the appropriate Virtual machine in the vCenter server resource pane.



Create Virtual Machine Backup

Virtual Machine

Some datastores are not on Isilon clusters. Only datastores on Isilon clusters will be used to create snapshots

Properties

Name:

ESX Host: 10.20.28.220

Data Sources: Local Storage /

Backup Settings

*Backup Name:

Comments:

Before creating the backup:

○ Shutdown the VM (COLD)

○ Pause the VM (WARM)

● Quiesce the VM by taking a temporary snapshot (HOT)

Within the ‘Create Virtual Machine Backup’ view, the ‘Properties’ field clearly identifies what data is covered, while the ‘Backup Settings’ fields allow a variety of machine quiescence types to be selected. As described above, choices depend on the backup requirements of the particular virtual machine and application versus the acceptable I/O pause interval.

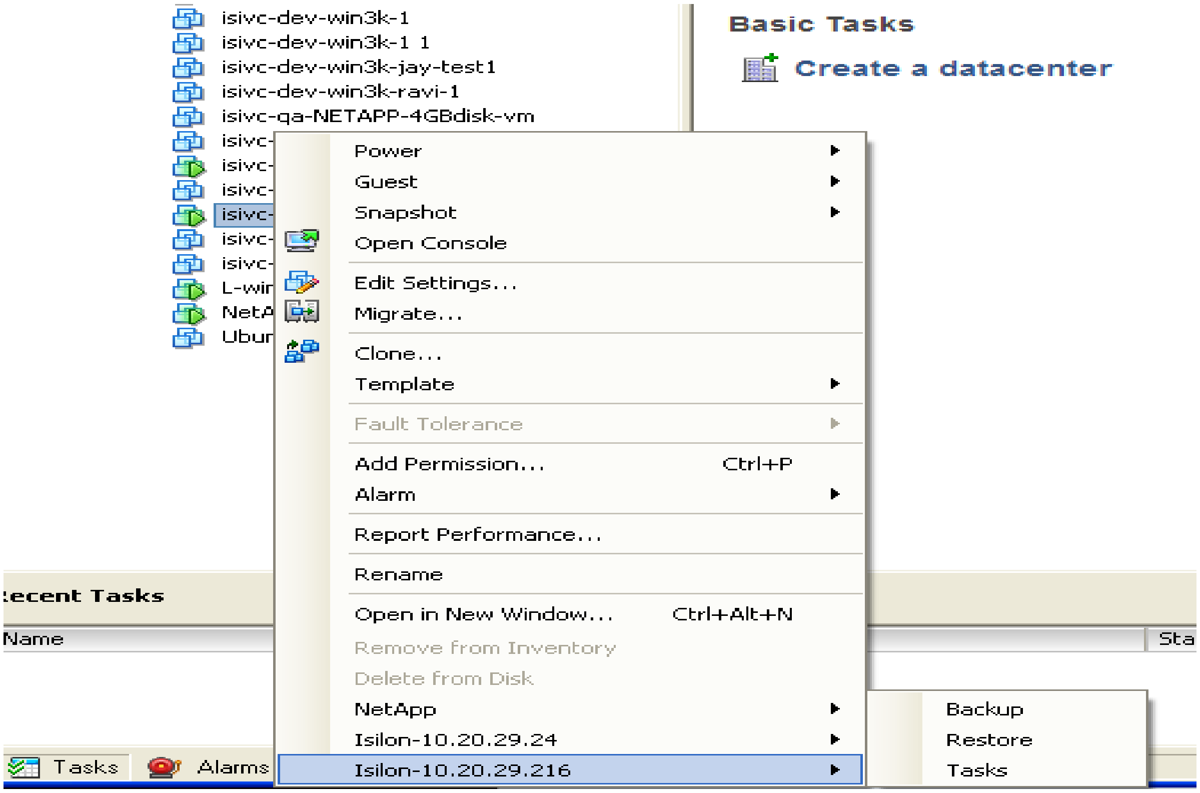
Note:

- In order for full application consistency (for instance, when backing up a database server), a ‘cold backup’ should always be selected.
- ‘Hot backup’ is the default selection.
- RDM snapshot and restore functionality is not available in this version of Isilon for vCenter.
- Isilon SnapshotIQ must be licensed and enabled in order to use the Isilon for vCenter functionality.

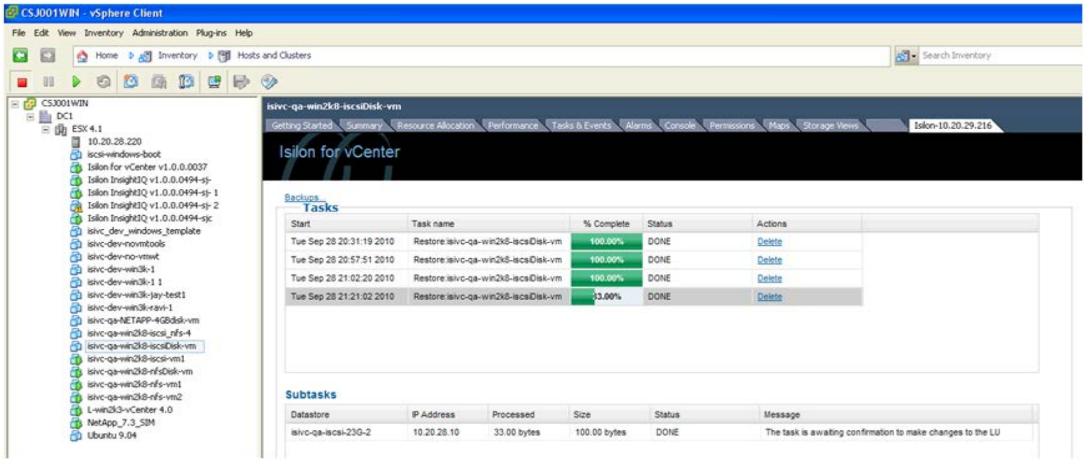
Restoring from a Previous Isilon for vCenter Backup.

In order to restore a virtual machine’s state to a certain point in time backup, the following procedure should be used:

1. From the vCenter server resource menu, right click on the desired virtual machine icon and the available, time-stamped snapshot objects for that VM are listed.



2. Select the appropriate point-in-time backup from the list and select ‘restore’ from the menu.
3. The following progress tracking screen will be displayed.



Tasks

Start	Task name	% Complete	Status	Actions
Tue Sep 28 20:31:19 2010	Restore:isivc-qa-win2k8-iscsiDisk-vm	100.00%	DONE	Delete
Tue Sep 28 20:57:51 2010	Restore:isivc-qa-win2k8-iscsiDisk-vm	100.00%	DONE	Delete
Tue Sep 28 21:02:20 2010	Restore:isivc-qa-win2k8-iscsiDisk-vm	100.00%	DONE	Delete
Tue Sep 28 21:21:02 2010	Restore:isivc-qa-win2k8-iscsiDisk-vm	33.00%	DONE	Delete

Subtasks

Datastore	IP Address	Processed	Size	Status	Message
isivc-qa-iscsi-23G-2	10.20.28.10	33.00 bytes	100.00 bytes	DONE	The task is awaiting confirmation to make changes to the LU

Note: This technique is mostly suited for NFS datastores where each VM is housed in its own directory on the Isilon cluster or iSCSI environments where there is a single virtual machine to LUN relationship. In these instances an Isilon IQ snapshot can be taken for individual VMs. For iSCSI datastores housing multiple virtual machines an Isilon IQ snapshot can only be taken on the entire datastore. In this case, all VMs on a shared iSCSI datastore must be quiesced before the snapshot is taken.

For more information on deploying Isilon for vCenter please refer to the ***Isilon for vCenter Install Guide*.**

Note: Additional snapshot information is also available in the following ***VMware knowledge base article.***

11. Disaster Recovery with SyncIQ

Using Isilon SyncIQ, virtual machines can be replicated to a secondary Isilon cluster and ESX host for disaster recovery, testing or other uses. Similar to Isilon IQ snapshots, the granularity of the replication data set depends on whether virtual machines are stored in iSCSI VMFS or NFS datastores:

1. NFS datastores allow replicating individual virtual machines because each VM is stored in a separate directory on the Isilon cluster.
2. iSCSI VMFS datastores map to iSCSI LUN directories on the Isilon cluster so all VMs in that LUN are replicated at once.

SyncIQ uses Isilon IQ snapshots to asynchronously replicate a point-in-time view of VM directories and LUNs between clusters and scheduling can be automated by creating SyncIQ replication policies. SyncIQ optimizes data transfer over the network by replicating only changed blocks between subsequent replication events.

On the target SyncIQ replication cluster Isilon IQ snapshots can be taken after each replication event to keep multiple version of each replication and revert back in time to the selected version.

Restoring virtual machines from the target replication target can be done in several ways:

1. Stopping all replications to the target cluster and mounting the target cluster nodes as datastores. Once datastores are mounted VMs stored on those datastores can be registered and powered on by the same or remote ESX hosts.
2. Copying VMs or LUNs from the target cluster to another location locally. In this way the target cluster can continue to operate as a SyncIQ replication target while also servicing ESX host datastore storage.

3. Copying VMs from or LUNs from the target cluster back to the primary cluster and re-register those VMs by the original ESX host. The target cluster continues to operation only as a SyncIQ replication target (or serving a separate ESX set of hosts).

Note: Isilon SyncIQ does not provide VM file level restore. Please contact one of the following Isilon partners in virtualization data protection solutions to address file level restores: Commvault, Symantec, Vizioncore, Veeam.

Restoring Virtual Machines from SyncIQ Replication Sets

Once replicated, the remote cluster can be restored and accessed from the same ESX host or a secondary ESX host:

1. In the vSphere client, select the primary or secondary ESX host.
2. On the Summary page, double-click on the datastore residing on the secondary cluster to browse the datastore.
3. Navigate to the directory where the replicated VM resides.
4. Right-click on the VM's configuration file (.vmx file), then select Add to Inventory.
5. When the VM has been added to the ESX host's inventory, the VM can be powered on normally.

Note: When the VM is powered on, a dialog may appear indicating that a duplicate uuid was found. The uuid is a unique machine identifier that gets copied with the VM configuration. If you want to use the original VM on the primary ESX host/Isilon cluster, then create a new VM UUID. This step can be automated by selecting Always Create.

Performance Tuning for SyncIQ Replication Jobs

The following options can improve SyncIQ replication jobs.

- Filtering out .vswp files from the replication set. .vswp files or virtual machines files used to store swapped out memory data from virtual machines. They represent run time data not required when restoring a virtual machine from a target replication set. Often times .vswp files comprise the majority of changed data between replication jobs adding considerable amount of data to transfer to the remote target. .vswp files can be filtered out of the replication set by using the filtering options available when defining a SyncIQ replication policy.
- Disabling block hash comparisons during replication jobs. By default SyncIQ jobs compare every block of existing files between the source and target replication sets to ensure they were note changed between replication jobs. This can significantly increase replication time and add considerable CPU load on the nodes conducting the hash calculations (disrupting other storage activity), particularly in environments where there is a fast network link between the source and remote cluster and CPU load will slow down network transfer. It is recommended to disable hash calculations by using the following command line on the source cluster where the SyncIQ policy is defined:

```
'isi sync policy modify --policy=<policy name> --skip_bb_hash=on'
```

For more information on SyncIQ for Isilon IQ replication please refer to the ***SyncIQ Best Practices Guide***.

12. Performing Backups

Most enterprises include tape backup as part of their data protection process. Tape backup is an important element of the picture; however, faster methods to restore lost data, such as snapshots and remote synchronization of data, are a first line of defense. For Isilon recommendations and best practices for data protection please see the ***Data Protection for Isilon IQ Scale-out Storage Best Practice Guide***.

In general, there are three options for backing up virtual machines:

1. Guest OS VM backups using backup client software (e.g. Commvault Galaxy or Symantec NetBackup) installed on each VM. Guest level backups using VM-installed software have the primary advantages of application awareness. The main disadvantage of this method is that as the number of virtual machines grows, resource utilization and scheduling of a growing number of backup agents become complex to manage.
2. vStorage API for Data Protection (VADP), formerly known as VMware Consolidated Backup (VCB), which centralizes the management of VM backups and integrates with leading data protection vendors, to provide additional flexibility in how virtual machines are backed up and restored.
3. Network Data Management Protocol (NDMP) backup from a NAS device using NDMP-compliant backup software. While NDMP backups are prevalent in NAS environments they do not provide application consistent snapshots of the virtual machines if such snapshots are required.

vStorage API for Data Protection (VADP)

With support for both NAS and iSCSI datastores, VADP is available for backups in an Isilon environment. VADP can perform image-level backups of entire virtual machines or file-level backups of Windows and Linux virtual machines. VADP is supported by most backup applications, allowing a familiar interface to be used for managing backups. VADP also utilizes VMware snapshot and Windows VSS integration to ensure data consistency with applications running inside Windows virtual machines.

For more details on this methodology, please the VMware website for VADP: http://www.vmware.com/products/vstorage-apis-for-data-protection/overview.html

Since all virtual machines are represented by files on the NFS datastore, using NDMP is a viable option. In this use case, all VM virtual disks, config files and snapshots (if they exist) are backed up using NDMP-compliant backup software. This method has the further advantage of taking the backup load off the ESX server machine(s). The main limitation of this method is similar to that of taking snapshots, i.e. VM and application state must be considered, and to backup a consistent copy of the VM, its applications and data, the VM should be shut down or suspended.

NDMP Backup of Virtual Machines

As noted, NDMP based backups only take crash consistent snapshots of the virtual machines and do not provide application consistent snapshots (such as databases driven applications).

NDMP backups are managed by backup software certified for compatibility with Isilon clustered storage. Isilon is certified with these products:

- Symantec NetBackup
- Commvault Simpana
- EMC Networker
- IBM Tivoli Storage Manager
- BakBone NetVault
- Atempo Time Navigator

For current information on certified OneFS and NDMP backup software versions, please consult the Isilon NDMP compatibility guide: http://www.isilon.com/pdfs/guides/Isilon_NDMP_Compatibility_Guide.pdf

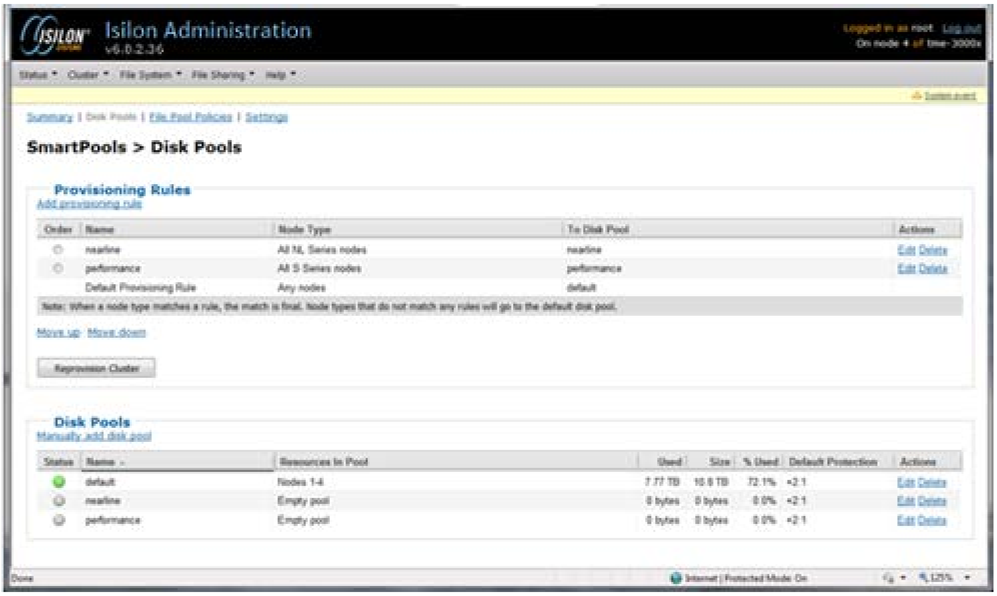
13. Managing Virtual Machine Performance and Protection

Isilon SmartPools

The combination of Isilon's SmartPools technology and VMware vSphere4 enables real-time or policy driven management of a virtual machine's performance. The SmartPools product applies storage tiering concepts to disk pools, facilitating storage alignment according to file policies or attributes. For example, virtual machines with a high IO requirement may be placed in a fast storage pool, whereas less subscribed virtual machines can reside in a less expensive pool.

Isilon SmartPools includes the following features:

1. Disk pools: Dynamic groups of disks associated in a single pool of storage, for example "all disks of all S-series nodes on the cluster." Disk pool membership changes through the addition or removal of nodes and drives.
2. Disk pool provisioning: Rules to automatically allocate new hardware to disk pools as it is added.



Isilon Administration
v6.0.2.36
Logged in as root Log out
On node 4 of tme-3000x
Status ▾ Cluster ▾ File System ▾ File Sharing ▾ Help ▾
System event
Summary | Disk Pools | File Pool Policies | Settings

SmartPools > Disk Pools

Provisioning Rules

Add provisioning rule

Order	Name	Node Type	To Disk Pool	Actions
○	nearline	All NL Series nodes	nearline	Edit Delete
○	performance	All S Series nodes	performance	Edit Delete
	Default Provisioning Rule	Any nodes	default	

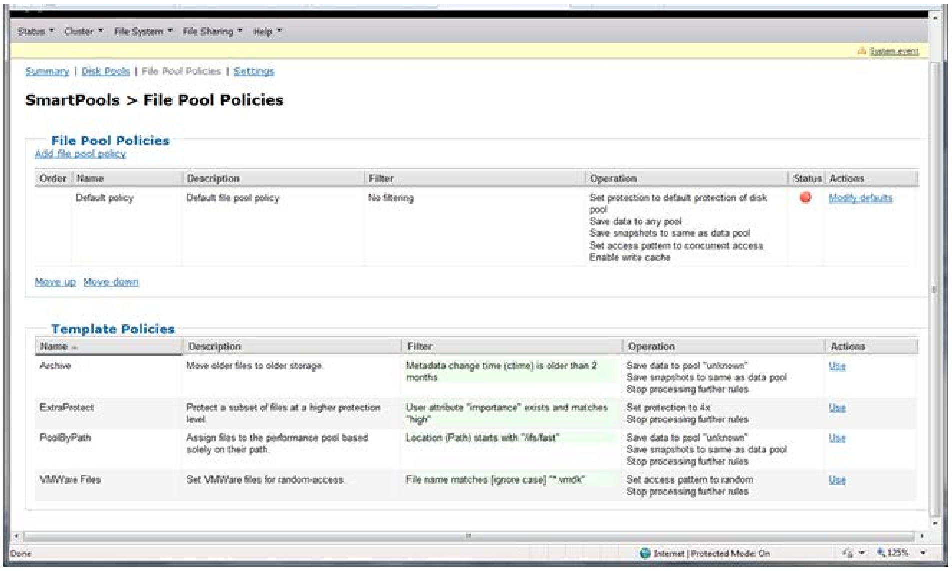
Note: When a node type matches a rule, the match is final. Node types that do not match any rules will go to the default disk pool.

Disk Pools

Manually add disk pool

Status	Name	Resources In Pool	Used	Size	% Used	Default Protection	Actions
●	default	Nodes 1-4	7.77 TB	10.8 TB	72.1%	+2:1	Edit Delete
○	nearline	Empty pool	0 bytes	0 bytes	0.0%	+2:1	Edit Delete
○	performance	Empty pool	0 bytes	0 bytes	0.0%	+2:1	Edit Delete

3. Virtual hot spares: Reserved space in a disk pool (up to four full drives) which can be used for data re-protection in the event of a drive failure.
4. File pools: Logical collections of files and associated policies governing attributes such as file size, file type, location, and file creation, change, modification, and access times.



Status ▾ Cluster ▾ File System ▾ File Sharing ▾ Help ▾
System event
Summary | Disk Pools | File Pool Policies | Settings

SmartPools > File Pool Policies

File Pool Policies

Add file pool policy

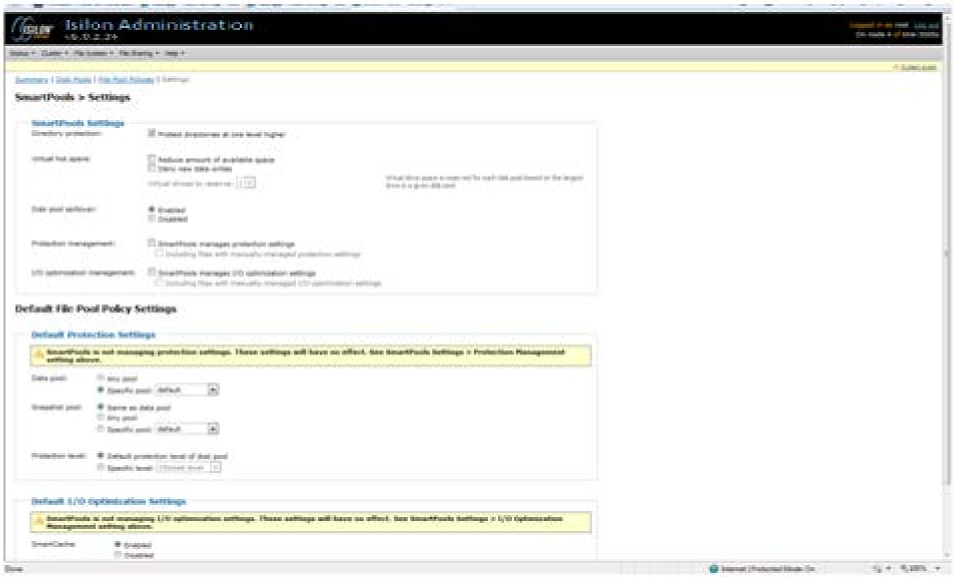
Order	Name	Description	Filter	Operation	Status	Actions
	Default policy	Default file pool policy	No filtering	Set protection to default protection of disk pool Save data to any pool Save snapshots to same as data pool Set access pattern to concurrent access Enable write cache	●	Modify defaults

Move up Move down

Template Policies

Name	Description	Filter	Operation	Actions
Archive	Move older files to older storage.	Metadata change time (ctime) is older than 2 months	Save data to pool "unknown" Save snapshots to same as data pool Stop processing further rules	Use
ExtraProtect	Protect a subset of files at a higher protection level.	User attribute "importance" exists and matches "high"	Set protection to 4x Stop processing further rules	Use
PoolByPath	Assign files to the performance pool based solely on their path.	Location (Path) starts with "/ifs/fast"	Save data to pool "unknown" Save snapshots to same as data pool Stop processing further rules	Use
VMWare Files	Set VMWare files for random-access.	File name matches [ignore case] "*.vmdk"	Set access pattern to random Stop processing further rules	Use

5. Disk pool spillover management: Rules governing handling of write operations to a full disk pool.



Isilon Administration
[illegible]
Status ▾ Cluster ▾ File System ▾ File Sharing ▾ Help ▾
Summary | Disk Pools | File Pool Policies | Settings

SmartPools > Settings

SmartPools Settings

Directory protection: ☑ Protect directories at one level higher

Virtual hot spare: □ Reduce amount of available space
□ Deny new data writes
Virtual drives to reserve: [illegible]
[illegible]

Disk pool spillover: ● Enabled
○ Disabled

Protection management: □ SmartPools manages protection settings
□ Including files with manually-managed protection settings

I/O optimization management: □ SmartPools manages I/O optimization settings
□ Including files with manually-managed I/O optimization settings

Default File Pool Policy Settings

Default Protection Settings

SmartPools is not managing protection settings. These settings will have no effect. See SmartPools Settings > Protection Management setting above.

Data pool: ○ Any pool
● Specific pool: default

Snapshot pool: ● Same as data pool
○ Any pool
○ Specific pool: default

Protection level: ● Default protection level of disk pool
○ Specific level: [illegible]

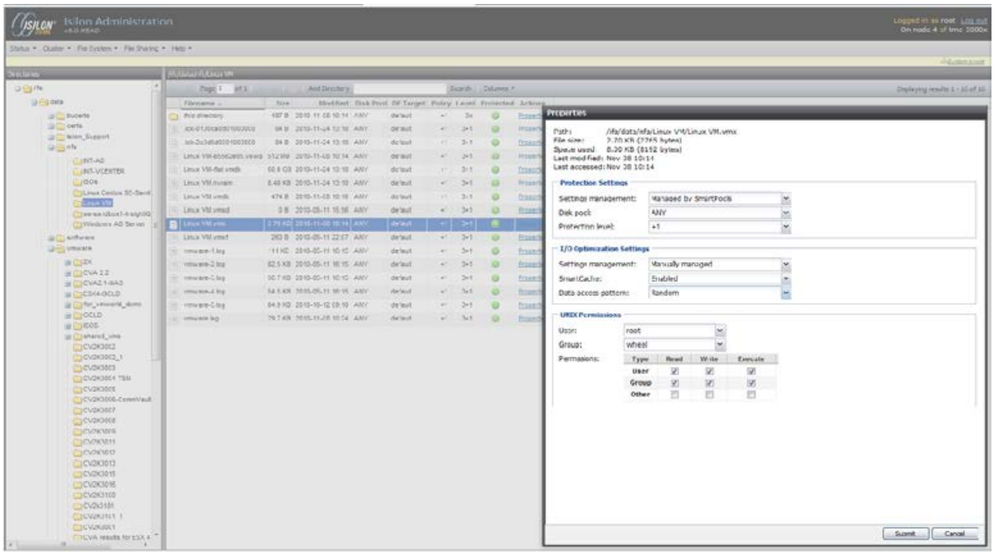
Default I/O Optimization Settings

SmartPools is not managing I/O optimization settings. These settings will have no effect. See SmartPools Settings > I/O Optimization Management setting above.

SmartCache: ● Enabled
○ Disabled

Using Isilon SmartPools it is possible to dynamically align virtual machine files, datastores and RDMs. These can be placed on the appropriate class of storage (SSD, SAS or SATA) via disk pools.

And with SmartPools file pools, protection (parity or mirroring) and IO optimization can be applied per-file, per-VM or per-Datastore.



Properties

Path: /ifs/data/ifs/Linux VM/Linux VM.vmx
File size: 2.70 KB (2765 bytes)
Space used: 8.30 KB (8152 bytes)
Last modified: Nov 30 10:14
Last accessed: Nov 30 10:14

Protection Settings

Settings management: Managed by SmartPools
Disk pool: ANY
Protection level: +1

I/O Optimization Settings

Settings management: Manually managed
SmartCache: Enabled
Data access pattern: Random

UNIX Permissions

User: root
Group: wheel
Permissions:

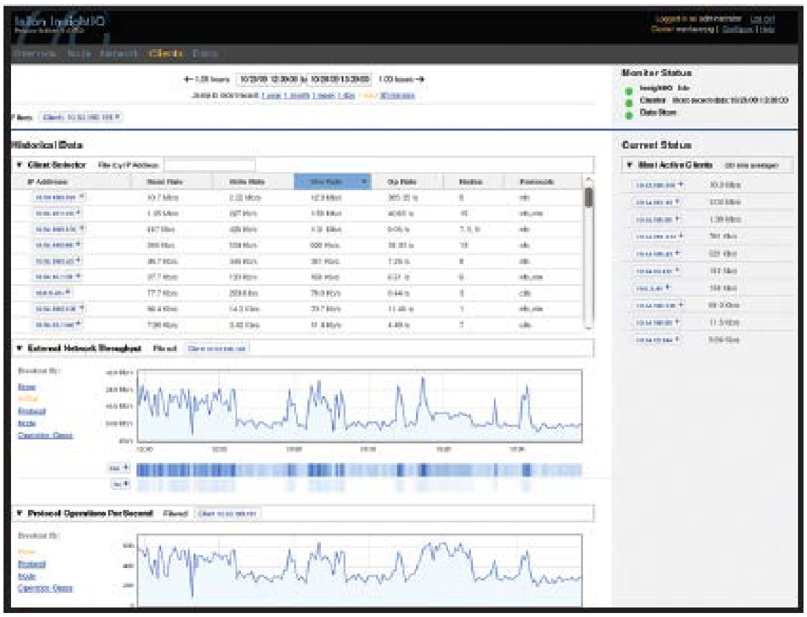
Type	Read	Write	Execute
User	☑	☑	☑
Group	☑	☑	☑
Other	☐	☐	☐

Submit Cancel

For more information on SmartPools for Isilon IQ, please refer to the ***SmartPools Best Practices Guide.***

Isilon InsightIQ

In conjunction with SmartPools, Isilon OneFS 6.0 also features the InsightIQ analytics product. Insight IQ is a real-time and historical performance trending and analysis utility, affording introspective metrics views at a per-virtual machine or per-file granularity. Built on a Web 2.0 framework, Insight IQ is deployed as a virtual appliance and provides vital diagnostic and workflow optimization data for managing virtual environments.



For more information on Isilon InsightIQ, please refer to the ***InsightIQ Best Practices Guide.***

14. Virtual Machine Storage Resource Balancing

Storage IO Control

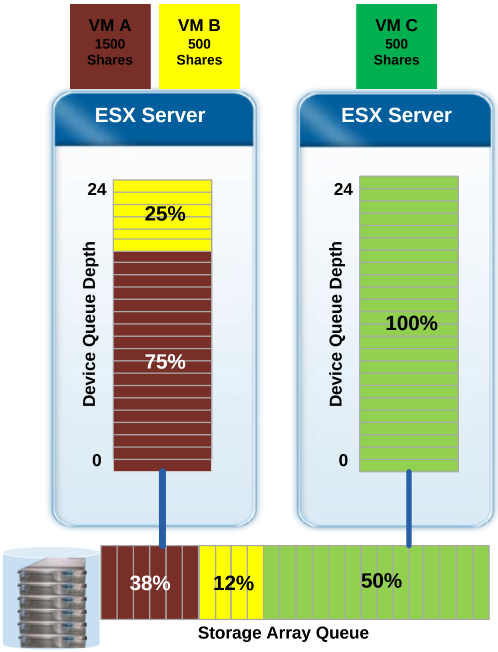
With vSphere4.1, VMware introduced the Storage IO Control (SIOC) functionality.

Using SIOC, it is now possible to balance virtual machine IO across asymmetrically loaded ESX hosts. To achieve this, Storage IO Control uses an IO throttling mechanism across disk shares to grant each VM its fair share of resources, in terms of contention (latency).

Simply enabling Storage I/O Control on each datastore will prevent a single VM from monopolizing the storage resources by leveling out all requests for I/O that the datastore receives.

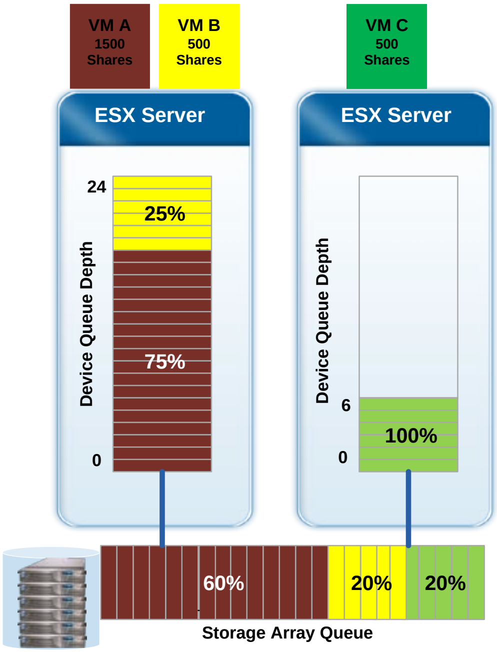
Without Storage I/O Control

Actual Disk Resources utilized by each VM are not in the correct ratio



With Storage I/O Control

Actual Disk Resources utilized by each VM are in the correct ratio event across ESX Hosts

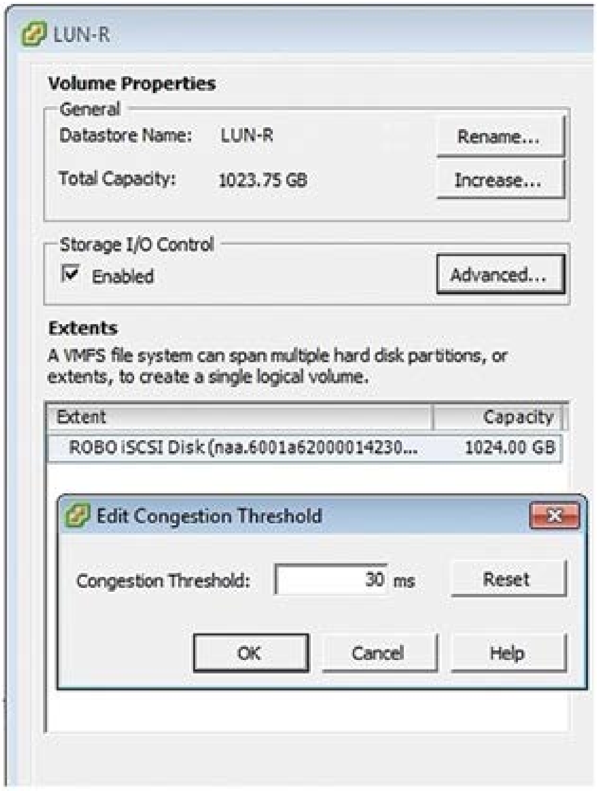


Note: Storage IO Control is disabled by default. It is set via vCenter on a per-datastore level and configured with a default latency trigger of 30ms for a sustained period of 4 seconds.

Configuring Storage IO Control

From within vCenter, configuring SIOC is a three-step process per datastore:

1. Enable SIOC for the desired datastore.
2. Set an appropriate congestion threshold.



LUN-R

Volume Properties

General

Datastore Name: LUN-R

Total Capacity: 1023.75 GB

Storage I/O Control

☑ Enabled

Extents

A VMFS file system can span multiple hard disk partitions, or extents, to create a single logical volume.

Extent	Capacity
ROBO iSCSI Disk (naa.6001a62000014230...	1024.00 GB

Edit Congestion Threshold

Congestion Threshold: 30 ms

Note: The Isilon recommendation for the congestion threshold value is 20ms (default is 30ms).

3. Configure the number of storage IO shares and the upper limit of IOPs for each virtual machine. By default, all virtual machines are set to ‘Normal’ (value 1000), and with unlimited IOPS.

Note: Currently Storage IO Control is only available for Virtual Machine File System (VMFS) volumes.

For further Storage IO Control configuration information, please refer to the ***VMware vSphere 4.1 vCenter product documentation.***

About Isilon: As the global leader in scale-out storage, Isilon delivers powerful yet simple solutions for enterprises that want to manage their data, not their storage. Isilon's products are simple to install, manage and scale, at any size. And, unlike traditional enterprise storage, Isilon stays simple no matter how much storage is added, how much performance is required or how business needs change in the future. Information about Isilon can be found at http://www.isilon.com.