5.4.4 iSCSI Facts

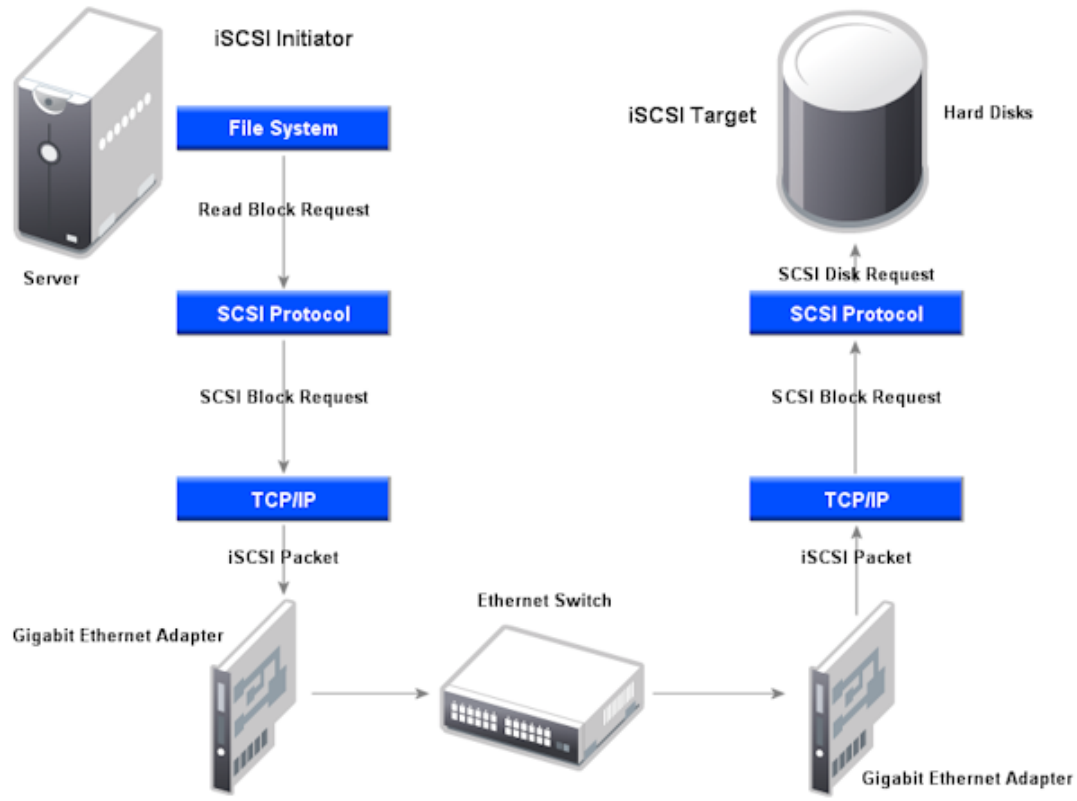
iSCSI is a network protocol that encapsulates SCSI block-level commands within IP packets for transmission over an Ethernet network. iSCSI uses Ethernet hardware to create the storage area network (SAN). Using standard Ethernet hardware lowers the cost of a SAN implementation when using iSCSI. The hardware required to create an iSCSI SAN includes the following:

- Ethernet cabling (fiber optic or Cat5/5e/6 UTP)
- Ethernet switches
- Ethernet NICs

Because iSCSI uses TCP/IP packets, the iSCSI SAN can be configured to run over the production network infrastructure with other networking devices. However, performance will likely suffer because the available bandwidth is shared by many devices. As an alternative, it is recommended that a second, parallel network infrastructure (cabling, switches, and NICs) be implemented that is dedicated only to the iSCSI SAN.

iSCSI storage devices are called *targets*. iSCSI servers use a logical entity called an iSCSI *initiator* to connect to and communicate with an iSCSI target. iSCSI uses a client/server connection between the iSCSI initiator (client) and iSCSI target (server) using port 3260 (by default). The storage devices on the remote iSCSI target appear to the operating system on the iSCSI initiator as locally-attached hard disks, effectively creating a SAN.

SCSI commands normally sent to locally-attached SCSI storage devices can also be sent over a network connection using the IP protocol. With iSCSI, the iSCSI initiator sends SCSI commands to the iSCSI target over the network. The iSCSI target system redirects the SCSI commands to a directly-attached storage device, as depicted in the following:



To implement and manage an iSCSI SAN, be familiar with the following iSCSI terminology:

- A *network entity* is a device connected to an IP network that hosts one or more iSCSI nodes.
- The *network portal* is the IP address and port through which an iSCSI node is accessed on a network entity.
- The *Protocol Data Unit* (PDU) is a data structure that carries the messages communicated between the iSCSI initiator and target.
- An *iSCSI node* can be either an iSCSI target or iSCSI initiator. A network entity can host one or more iSCSI nodes that are accessible through one or more network portals.
- The *iSCSI name* is the unique name of an iSCSI node, typically an *iSCSI Qualified Name* (IQN). Assigning unique names to iSCSI nodes allows multiple nodes to share the same network portal on a network entity.
- The *iSCSI Qualified Name* (IQN) is designed to be globally unique using Internet naming standards. It is comprised of the unique identifier assigned to an iSCSI node, using the following syntax: **iqn.** ***year- month. TLD. Internet_domain***. The *year-month* portion of the IQN identifies the year and month the Internet domain was registered. The *TLD* portion of the IQN specifies the top-level domain of the Internet domain (such as com, net, org, and so on). The *Internet_domain* portion of the IQN identifies the registered Internet domain name.
- An *iSCSI target* is equivalent to a traditional SCSI device.
- An *iSCSI initiator* is equivalent to a traditional SCSI host bus adapter.
- In terms of iSCSI, a *LUN* is equivalent to a traditional SCSI disk. An iSCSI target hosts one or more LUNs.

When choosing between iSCSI and other SAN technologies, such as Fibre Channel (FC), be aware of the following:

- iSCSI is cheaper and can be easier to implement than FC. FC requires specialized hardware and knowledge, while iSCSI can be implemented using standard Ethernet hardware and TCP/IP.
- iSCSI is currently not as fast as FC. Using Gigabit Ethernet hardware, an iSCSI implementation can approximate the speed of the slowest FC SAN.
- Fibre Channel has a distance limitation of 10km. With iSCSI, IP packets can be sent over longer distances, if needed.
- iSCSI devices have built-in security features that FC devices do not. For example, Challenge Handshake Authentication Protocol (CHAP) can be enabled for authentication and IPsec for encryption. FC security is implemented primarily through limiting physical access.

To configure iSCSI:

- Create the SAN network infrastructure using Ethernet cables and switches.
- Attach the servers and their storage devices to the iSCSI SAN network using Ethernet network adapters.
- Configure iSCSI targets. The target identifies the storage device that iSCSI initiators will connect to and use over the SAN fabric:
  - Install the iSCSI Target Server Role.
  - Use the **iSCSI** option under File and Storage Services in Server Manager to manage iSCSI targets.
  - Create an iSCSI virtual disk on the iSCSI target server.
  - Define an iSCSI target on the server.
  - Specify the iSCSI initiators that are allowed to access the target.
  - Configure security settings for the target, such as the use of mutual CHAP or IPsec.
- Configure iSCSI initiators:
  - Start the iSCSI initiator service.
  - Specify the iSCSI target to connect to.
  - Configure security settings required by the target, such as the use of mutual CHAP or IPsec.
  - Bring the remote target disk online.
  - If necessary, create a volume on the disk.

    To configure the iSCSI initiator to automatically reconnect to the iSCSI target every time the system boots, add the iSCSI target to the Favorite Targets list in the iSCSI initiator.

Use **DiskRAID.exe** to:

- Create LUNs and new volumes (simple, striped, mirrored, RAID-5 volumes)
- Create iSCSI targets
- Extend or add a mirror to an existing volume
- View information about the iSCSI initiator and LUNs
- Repair fault-tolerant LUNs

Multipath I/O (MPIO) is a feature that provides support for using multiple data paths to a storage device on Windows. Configuring multiple paths between the server and the storage device provides increased performance for data access and ensures that data is accessible if a single component in the SAN fabric fails. Use **mpclaim** to manage MPIO. To use MPIO:

- Install multiple host bus adapters (HBAs) in the server and use SAN hardware (switches and cables) to create redundant paths between devices.
- Add the MPIO feature through Server Manager.
- Define MPIO policies to configure how load balancing and failover are performed.
