MODULE 3

IP SAN and FCoE

- Traditional SAN enables the transfer of block I/O over Fibre Channel and provides high performance and scalability.
- Advancements in technology have enabled IP to be used for transporting block I/O over the IP network. This technology of transporting block I/Os over an IP is referred to as IP SAN. With IP SAN, organizations can extend the geographical reach of their storage infrastructure
- Two primary protocols that leverage IP as the transport mechanism are Internet SCSI (iSCSI) and Fibre Channel over IP (FCIP).
- iSCSI is encapsulation of SCSI I/O over IP. FCIP is a protocol in which an FCIP entity such as an FCIP gateway is used to tunnel FC fabrics through an IP network.

6.1 iSCSI

- iSCSI is an IP based protocol that establishes and manages connections between host and storage over IP, as shown in Fig 6-1.
- iSCSI encapsulates SCSI commands and data into an IP packet and transports them using TCP/IP.
- iSCSI is widely adopted for connecting servers to storage because it is relatively inexpensive and easy to implement, especially in environments in which an FC SAN does not exist.

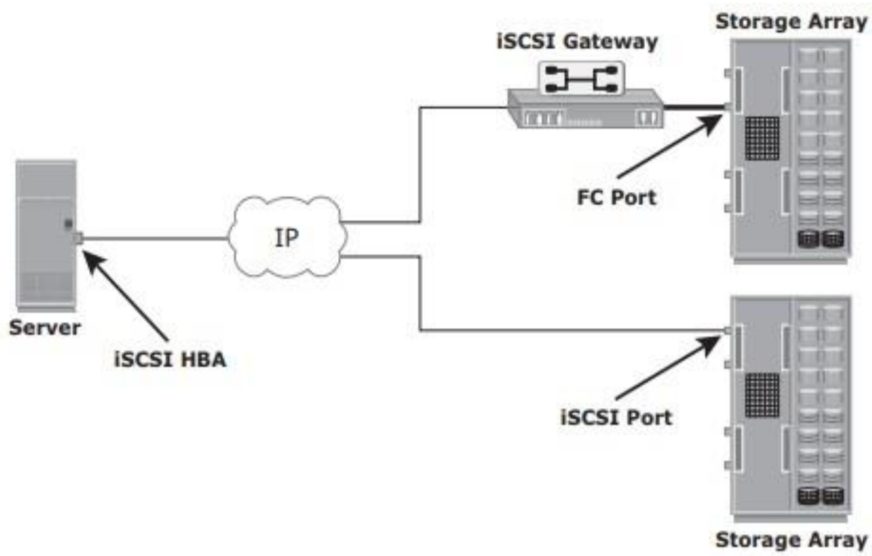

Fig 6-1: iSCSI implementation

6.1.1 Components of iSCSI

- An initiator (host), target (storage or iSCSI gateway), and an IP-based network are the key iSCSI components.
- If an iSCSI-capable storage array is deployed, then a host with the iSCSI initiator can directly communicate with the storage array over an IP network.
- However, in an implementation that uses an existing FC array for iSCSI communication, an iSCSI gateway is used.
- These devices perform the translation of IP packets to FC frames and vice versa, thereby bridging the connectivity between the IP and FC environments.

6.1.2 iSCSI Host Connectivity

The three iSCSI host connectivity options are:

- A standard NIC with software iSCSI initiator,
- a TCP offload engine (TOE) NIC with software iSCSI initiator,

- an iSCSI HBA

- The function of the iSCSI initiator is to route the SCSI commands over an IP network.
- A **standard NIC with a software iSCSI** initiator is the simplest and least expensive connectivity option. It is easy to implement because most servers come with at least one, and in many cases two, embedded NICs. It requires only a software initiator for iSCSI functionality. Because NICs provide standard IP function, encapsulation of SCSI into IP packets and decapsulation are carried out by the host CPU. This places additional overhead on the host CPU. If a standard NIC is used in heavy I/O load situations, the host CPU might become a bottleneck. TOE NIC helps reduce this burden.
- A **TOE NIC** offloads TCP management functions from the host and leaves only the iSCSI functionality to the host processor. The host passes the iSCSI information to the TOE card, and the TOE card sends the information to the destination using TCP/IP. Although this solution improves performance, the iSCSI functionality is still handled by a software initiator that requires host CPU cycles.
- An **iSCSI HBA** is capable of providing performance benefits because it offloads the entire iSCSI and TCP/IP processing from the host processor. The use of an iSCSI HBA is also the simplest way to boot hosts from a SAN environment via iSCSI. If there is no iSCSI HBA, modifi cations must be made to the basic operating system to boot a host from the storage devices because the NIC needs to obtain an IP address before the operating system loads. The functionality of an iSCSI HBA is similar to the functionality of an FC HBA.

6.1.3 iSCSI Topologies

- Two topologies of iSCSI implementations are **native and bridged**.
- Native topology does not have FC components.
- The initiators may be either directly attached to targets or connected through the IP network.
- Bridged topology enables the coexistence of FC with IP by providing iSCSI-to-FC bridging functionality.
- For example, the initiators can exist in an IP environment while the storage remains in an FC

environment.

Native iSCSI Connectivity

- FC components are not required for iSCSI connectivity if an iSCSI-enabled array is deployed.
- In Fig 6-2(a), the array has one or more iSCSI ports configured with an IP address and is connected to a standard Ethernet switch.
- After an initiator is logged on to the network, it can access the available LUNs on the storage array.
- A single array port can service multiple hosts or initiators as long as the array port can handle the amount of storage traffic that the hosts generate.

Bridged iSCSI Connectivity

- A bridged iSCSI implementation includes FC components in its configuration.
- Fig 6-2(b), illustrates iSCSI host connectivity to an FC storage array. In this case, the array does not have any iSCSI ports. Therefore, an external device, called a gateway or a multiprotocol router, must be used to facilitate the communication between the iSCSI host and FC storage.
- The gateway converts IP packets to FC frames and vice versa.
- The bridge devices contain both FC and Ethernet ports to facilitate the communication between the FC and IP environments.
- In a bridged iSCSI implementation, the iSCSI initiator is configured with the gateway's IP address as its target destination.
- On the other side, the gateway is configured as an FC initiator to the storage array.
- **Combining FC and Native iSCSI Connectivity:** The most common topology is a

combination of FC and native iSCSI. Typically, a storage array comes with both FC and iSCSI ports that enable iSCSI and FC connectivity in the same environment, as shown in Fig 6-2(c).

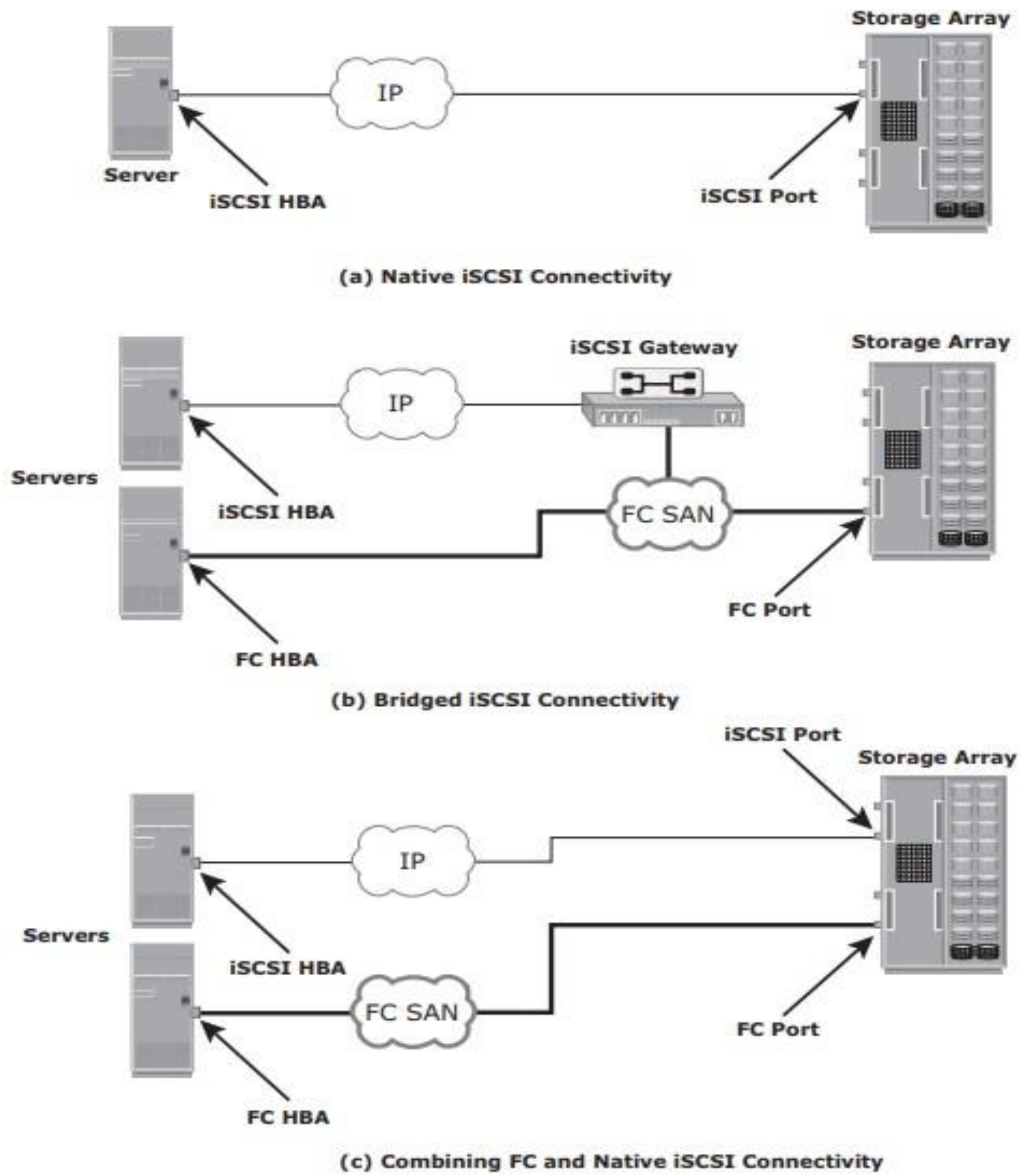

Fig 6-2: iSCSI Topologies

6.1.4 iSCSI Protocol Stack

- Fig 6-3 displays a model of the iSCSI protocol layers and depicts the encapsulation order of the SCSI commands for their delivery through a physical carrier.

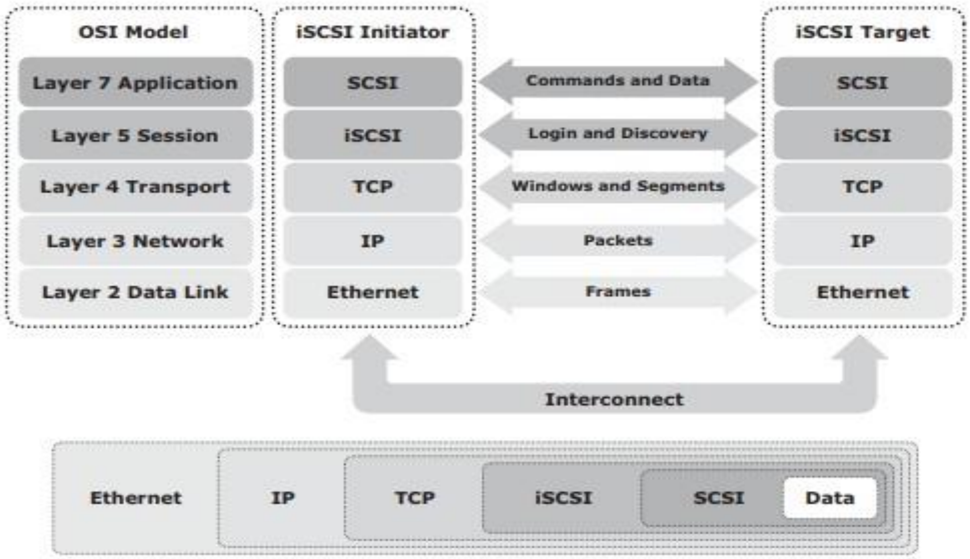

Fig 6-3: iSCSI protocol stack

- SCSI is the command protocol that works at the application layer of the Open System Interconnection (OSI) model.
- The initiators and targets use SCSI commands and responses to talk to each other.
- The SCSI command descriptor blocks, data, and status messages are encapsulated into TCP/IP and transmitted across the network between the initiators and targets.
- iSCSI is the session-layer protocol that initiates a reliable session between devices that recognize SCSI commands and TCP/IP.
- The iSCSI session-layer interface is responsible for handling login, authentication, target discovery, and session management.
- TCP is used with iSCSI at the transport layer to provide reliable transmission.

- TCP controls message flow, windowing, error recovery, and retransmission.
- It relies upon the network layer of the OSI model to provide global addressing and connectivity.
- The Layer 2 protocols at the data link layer of this model enable node-to-node communication through a physical network.

6.1.5 iSCSI PDU

- A *protocol data unit* (PDU) is the basic "information unit" in the iSCSI environment.
- The iSCSI initiators and targets communicate with each other using iSCSI PDUs. This communication includes establishing iSCSI connections and iSCSI sessions, performing iSCSI discovery, sending SCSI commands and data, and receiving SCSI status.
- All iSCSI PDUs contain one or more header segments followed by zero or more data segments.
- The PDU is then encapsulated into an IP packet to facilitate the transport.
- A PDU includes the components shown in Fig 6-4.
- The IP header provides packet-routing information to move the packet across a network.
- The TCP header contains the information required to guarantee the packet delivery to the target.
- The iSCSI header (basic header segment) describes how to extract SCSI commands and data for the target. iSCSI adds an optional CRC, known as the *digest*, to ensure datagram integrity. This is in addition to TCP checksum and Ethernet CRC. The header and the data digests are optionally used in the PDU to validate integrity and data placement.

As shown in Figure 6-5, each iSCSI PDU does not correspond in a 1:1 relationship with an IP packet. Depending on its size, an iSCSI PDU can span an IP packet or even coexist with another PDU in the same packet.

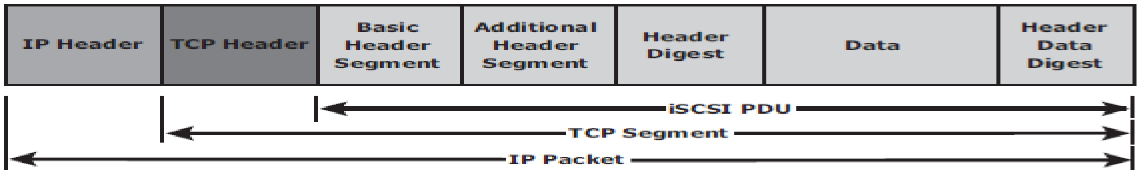

Fig 6-4: iSCSI PDU encapsulated in an IP packet

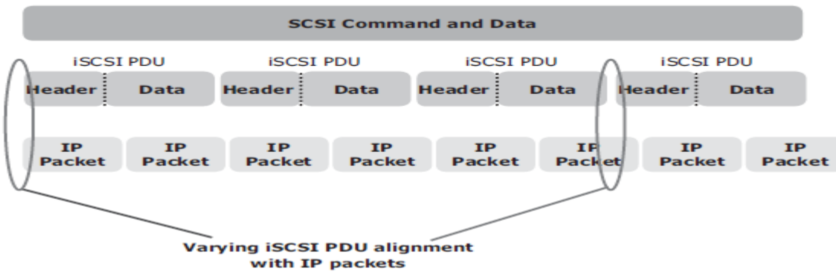

Figure 6-5: Alignment of iSCSI PDUs with IP packets

6.1.6 iSCSI Discovery

- An initiator must discover the location of its targets on the network and the names of the targets available to it before it can establish a session.
- This discovery can take place in two ways:
  - **SendTargets discovery**
  - **internet Storage Name Service (iSNS).**
- In *SendTargets discovery*, the initiator is manually configured with the target's network portal to establish a discovery session. The initiator issues the SendTargets command, and the target network portal responds with the names and addresses of the targets available to the host.
- iSNS (Fig 6-6) enables automatic discovery of iSCSI devices on an IP network. The initiators and targets can be configured to automatically register themselves with the iSNS server. Whenever an initiator wants to now the targets that it can access, it can query the iSNS server for a list of available targets.
- The discovery can also take place by using service location protocol (SLP). However, this is less commonly used than SendTargets discovery and iSNS.

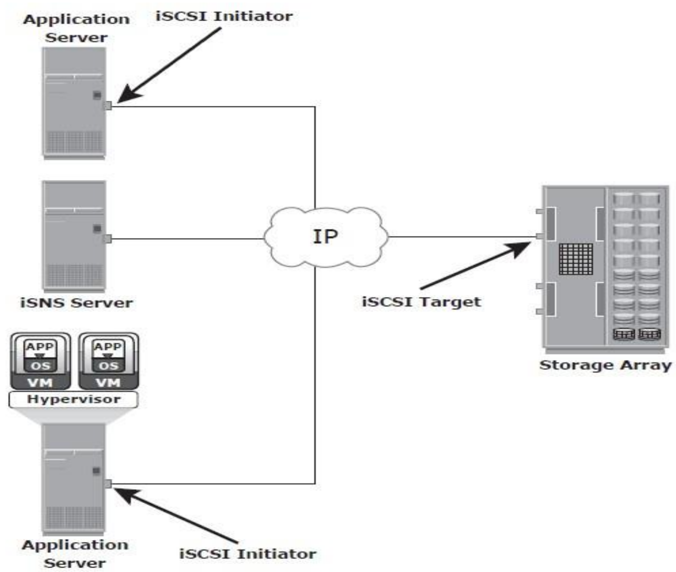

Fig 6-6: Discovery using iSNS

6.1.7 iSCSI Names

- A unique worldwide iSCSI identifier, known as an *iSCSI name*, is used to identify the initiators and targets within an iSCSI network to facilitate communication.
- The unique identifier can be a combination of the names of the department, application, or manufacturer, serial number, asset number, or any tag that can be used to recognize and manage the devices.
- Following are two types of iSCSI names commonly used:

    - **iSCSI Qualified Name (IQN):**
    - **Extended Unique Identifier (EUI)**

- **iSCSI Qualified Name (IQN):** An organization must own a registered domain name to generate iSCSI Qualifi ed Names. This domain name does not need to be active or resolve to an address. It just needs to be reserved to prevent other organizations from using the same domain name to generate iSCSI names. A date is included in the name to avoid potential conflicts caused by the transfer of domain names.

  An example of an IQN is iqn.2008-02.com.example:*optional_string*. The *optional_string* provides a serial number, an asset number, or any otherdevice identifiers.

- **Extended Unique Identifi er (EUI):** An EUI is a globally unique identifier based on the IEEE EUI-64 naming standard. An EUI is composed of the eui prefix followed by a 16-character hexadecimal name, such aseui.0300732A32598D26.

- In either format, the allowed special characters are dots, dashes, and blank spaces.

6.1.8 iSCSI Session

- An iSCSI session is established between an initiator and a target, as shown in Fig 6-7.
- A session is identified by a session ID (SSID), which includes part of an initiator ID and a target ID.
- The session can be intended for one of the following:
  - The discovery of the available targets by the initiators and the location of a specific target on a network
  - The normal operation of iSCSI (transferring data between initiators and targets)
- There might be one or more TCP connections within each session. Each TCP connection within the session has a unique connection ID (CID).
- An iSCSI session is established via the iSCSI login process. The login process is started when the initiator establishes a TCP connection with the required target either via the well-known port 3260 or a specified target port.

- During the login phase, the initiator and the target authenticate each other and negotiate on various parameters.
- After the login phase is successfully completed, the iSCSI session enters the full-feature phase for normal SCSI transactions. In this phase, the initiator may send SCSI commands and data to the various LUNs on the target.
- The final phase of the iSCSI session is the connection termination phase, which is referred to as the logout procedure.
- The initiator is responsible for commencing the logout procedure; however, the target may also prompt termination by sending an iSCSI message, indicating the occurrence of an internal error condition.
- After the logout request is sent from the initiator and accepted by the target, no further request and response can be sent on that connection.

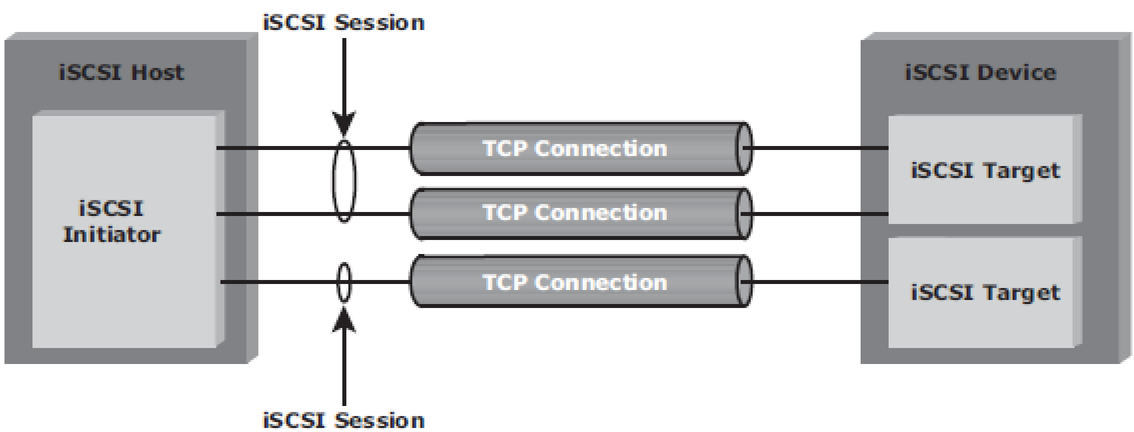

Fig 6-7: iSCSI session

6.1.9 Command Sequencing

- The iSCSI communication between the initiators and targets is based on the request-response command sequences.
- A command sequence may generate multiple PDUs.

- A ***command sequence number*** **(CmdSN**) within an iSCSI session is used for numbering all initiator-to-target command PDUs belonging to the session.
- This number ensures that every command is delivered in the same order in which it is transmitted, regardless of the TCP connection that carries the command in the session.
- Command sequencing begins with the first login command, and the CmdSN is incremented by one for each subsequent command.
- The iSCSI target layer is responsible for delivering the commands to the SCSI layer in the order of their CmdSN.
- Similar to command numbering, a ***status sequence number*** **(StatSN)** is used to sequentially number status responses, as shown in Fig 6-8.
- These unique numbers are established at the level of the TCP connection.
- A target sends ***request-to-transfer*** **(R2T)** PDUs to the initiator when it is ready to accept data.
- A *data sequence number* (DataSN) is used to ensure in-order delivery of data within the same command.
- The DataSN and R2TSN are used to sequence data PDUs and R2Ts, respectively.

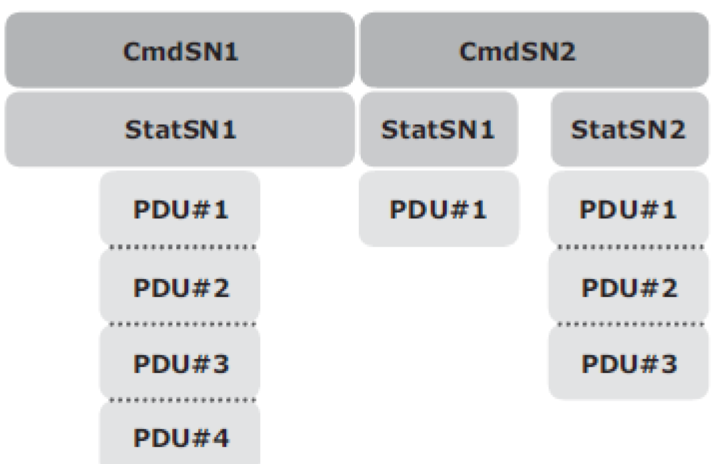

Fig 6-8: Command and status sequence number

6.2 FCIP (Fibre channel over IP)

- FCIP is a IP-based protocol that is used to connect distributed FC-SAN islands.
- Creates virtual FC links over existing IP network that is used to transport FC data between different FC SANS.
- It encapsulates FC frames into IP packet.
- It provides disaster recovery solution.

6.2.1 FCIP Protocol Stack

- The FCIP protocol stack is shown in Fig 6-9. Applications generate SCSI commands and data, which are processed by various layers of the protocol stack.
- The upper layer protocol SCSI includes the SCSI driver program that executes the read-and-write commands.
- Below the SCSI layer is the Fibre Channel Protocol (FCP) layer, which is simply a Fibre Channel frame whose payload is SCSI.
- The FCP layer rides on top of the Fibre Channel transport layer. This enables the FC frames to run natively within a SAN fabric environment. In addition, the FC frames can be encapsulated into the IP packet and sent to a remote SAN over the IP.

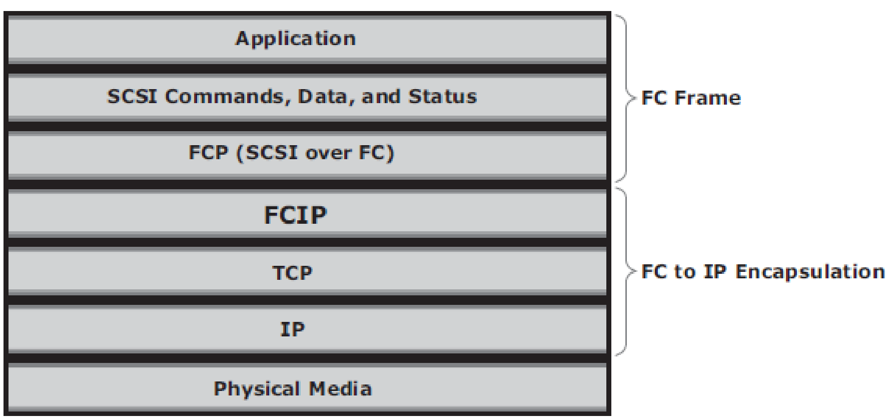

Fig 6-9: FCIP protocol stack

- The FCIP layer encapsulates the Fibre Channel frames onto the IP payload and passes them to the TCP layer (see Fig 6-10). TCP and IP are used for transporting the encapsulated information across Ethernet, wireless, or other media that support the TCP/IP traffic.
- Encapsulation of FC frame into an IP packet could cause the IP packet to be fragmented when the data link cannot support the maximum transmission unit (MTU) size of an IP packet.
- When an IP packet is fragmented, the required parts of the header must be copied by all fragments.
- When a TCP packet is segmented, normal TCP operations are responsible for receiving and re-sequencing the data prior to passing it on to the FC processing portion of the device.

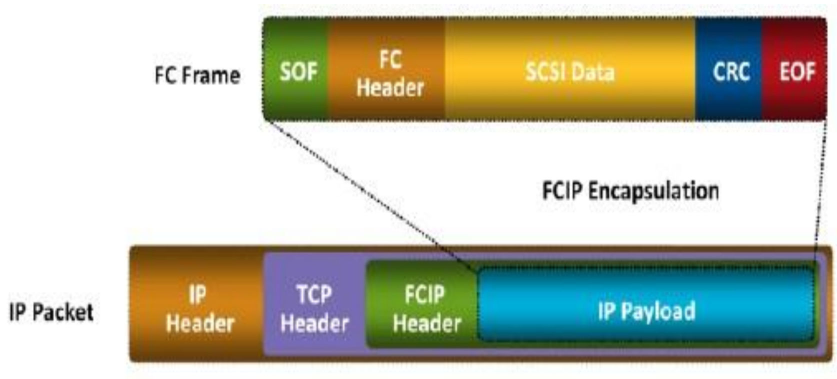

Fig 6-10: FCIP encapsulation

6.2.2 FCIP Topology

- In an FCIP environment, an FCIP gateway is connected to each fabric via a standard FC connection (Fig 6-11).
- The FCIP gateway at one end of the IP network encapsulates the FC frames into IP packets.
- The gateway at the other end removes the IP wrapper and sends the FC data to the layer 2 fabric.
- The fabric treats these gateways as layer 2 fabric switches.
- An IP address is assigned to the port on the gateway, which is connected to an IP network. After the IP connectivity is established, the nodes in the two independent fabrics can communicate

with each other.

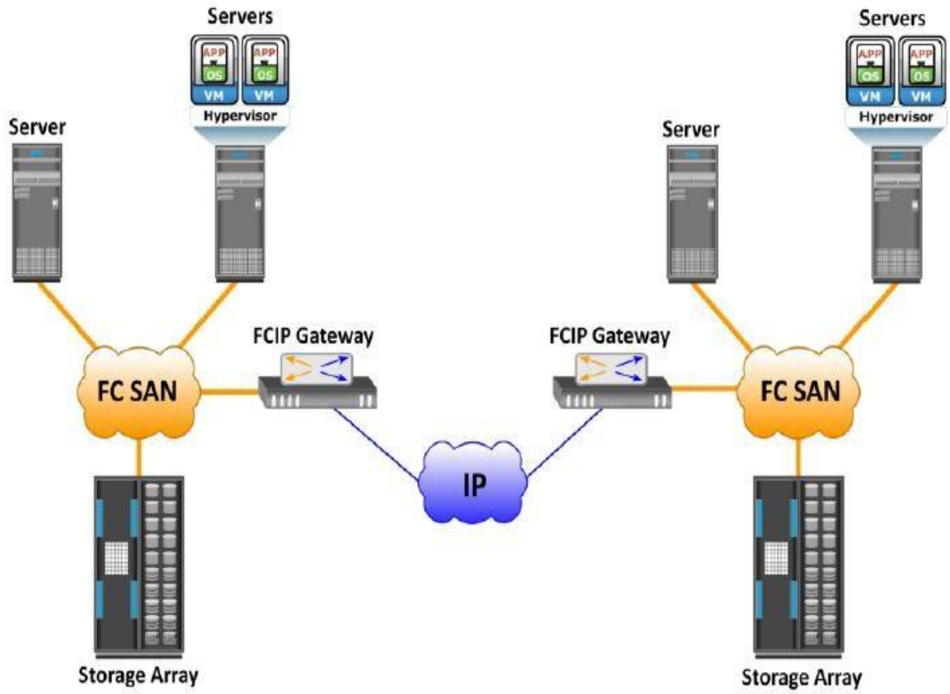

Fig 6-11: FCIP topology

6.2.3 FCIP Performance and Security

- Performance, reliability, and security should always be taken into consideration when implementing storage solutions. The implementation of FCIP is also subject to the same considerations.

- In addition, because FCIP creates a unified fabric, disruption in the underlying IP network can cause instabilities in the SAN environment.

- Security is also a consideration in an FCIP solution because the data is transmitted over public IP channels. Various security options are available to protect the data based on the router's support. IPSec is one such security measure that can be implemented in the FCIP environment.

CHAPTER 7 NETWORK ATTACHED STORAGE (NAS)

- File sharing enables users to share files with other users
- Network-based fi le sharing provides the flexibility to share files over long distances among a large number of users.
- File servers use client-server technology to enable fi le sharing over a network. To address the tremendous growth of file data in enterprise environments, organizations have been deploying large numbers of file servers.
- These servers are either connected to direct-attached storage (DAS) or storage area network (SAN)-attached storage. This has resulted in the proliferation of islands of over-utilized and under-utilized file servers and storage.
- In addition, such environments have poor scalability, higher management cost, and greater complexity. Network-attached storage (NAS) emerged as a solution to these challenges.

What is NAS?

- NAS is an IP based dedicated, high-performance file sharing and storage device.
- Enables NAS clients to share files over an IP network.
- Uses network and file-sharing protocols to provide access to the file data.
- Ex: Common Internet File System (CIFS) and Network File System (NFS).
- Enables both UNIX and Microsoft Windows users to share the same data seamlessly.
- NAS device uses its own operating system and integrated hardware and software components to meet specific file-service needs.
- Its operating system is optimized for file I/O which performs better than a general-purpose server.
- A NAS device can serve more clients than general-purpose servers and provide the benefit of server consolidation.

7.1 General-Purpose Servers versus NAS Devices

- A NAS device is optimized for file-serving functions such as storing, retrieving, and accessing fi les for applications and clients.
- As shown in Figure 7-1, a general-purpose server can be used to host any application because it runs a general-purpose operating system.
- Unlike a general-purpose server, a NAS device is dedicated to file-serving. It has specialized operating system dedicated to fi le serving by using industry-standard protocols.
- Some NAS vendors support features, such as native clustering for high availability

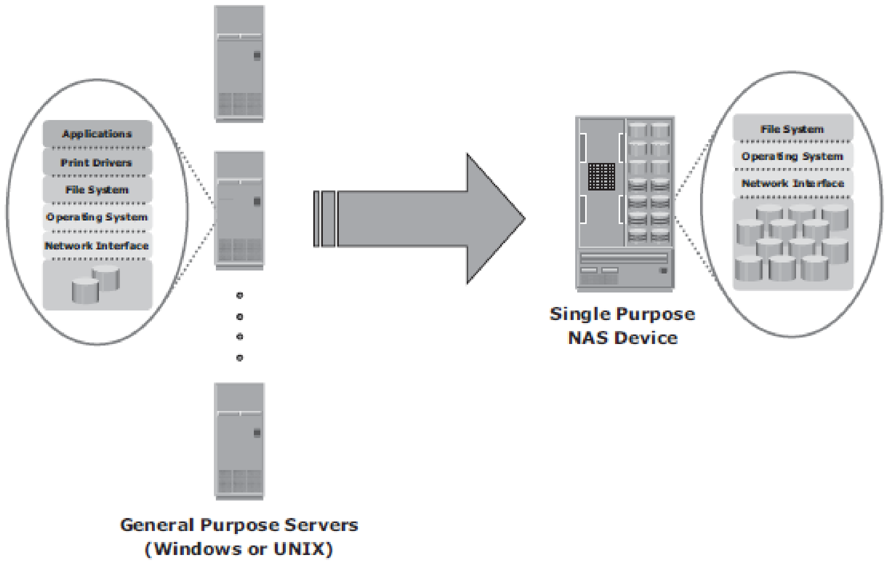

Figure 7-1: General purpose server versus NAS device

7.2Benefits of NAS

NAS offers the following benefits:

- ✓ **Comprehensive access to information:** Enables efficient fi le sharing and supports many-to-one and one-to-many configurations. The many-to-one configuration enables a NAS device to serve many clients simultaneously. The one-to-many configuration enables one client to connect with many NAS devices simultaneously.
- ✓ **Improved efficiency:** NAS delivers better performance compared to a general-purpose file server because NAS uses an operating system specialized for file serving.
- ✓ **Improved flexibility:** Compatible with clients on both UNIX and Windows platforms using industry-standard protocols. NAS is flexible and can serve requests from different types of clients from the same source.
- ✓ **Centralized storage:** Centralizes data storage to minimize data duplication on client workstations, and ensure greater data protection
- ✓ **Simplified management:** Provides a centralized console that makes it possible to manage file

systems efficiently

- ✓ **Scalability:** Scales well with different utilization profiles and types of business applications because of the high-performance and low-latency design
- ✓ **High availability:** Offers efficient replication and recovery options, enabling high data availability. NAS uses redundant components that provide maximum connectivity options. A NAS device supports clustering technology for failover.
- ✓ **Security:** Ensures security, user authentication, and file locking with industry-standard security schemas
- ✓ **Low cost:** NAS uses commonly available and inexpensive Ethernet components.
- ✓ **Ease of deployment:** Configuration at the client is minimal, because the clients have required NAS connection software built in.

7.3File Systems and Network File Sharing

A file system is a structured way to store and organize data fi les. Many file systems maintain a file access table to simplify the process of searching and accessing files.

7.3.1 Accessing a File System

- ➢ A file system must be mounted before it can be used. In most cases, the operating system mounts a local file system during the boot process.
- ➢ The mount process creates a link between the fi le system on the NAS and the operating system on the client.
- ➢ When mounting a file system, the operating system organizes files and directories in a tree-like structure and grants the privilege to the user to access this structure. The tree is rooted at a mount point.
- ➢ The mount point is named using operating system conventions. Users and applications can traverse the entire tree from the root to the leaf nodes as file system permissions allow.
- ➢ Files are located at leaf nodes, and directories and subdirectories are located at intermediate

roots. The access to the file system terminates when the file system is unmounted.

- Figure 7-2 shows an example of a UNIX directory structure.

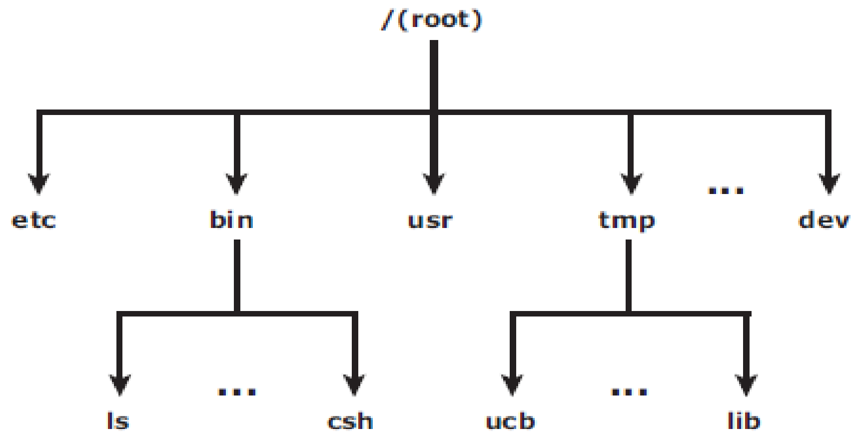

Figure 7-2: UNIX directory structure

7.3.2 Network File Sharing

Network file sharing refers to storing and accessing files over a network. In a file-sharing environment, the user who creates a file (the creator or owner of a file) determines the type of access (such as read, write, execute, append, and delete) to be given to other users and controls changes to the file.

- When multiple users try to access a shared file at the same time, a locking scheme is required to maintain data integrity and, at the same time, make this sharing possible.
- Some examples of file-sharing methods are file transfer protocol (FTP), Distributed File System (DFS), client-server models that use fi le-sharing protocols such as NFS and CIFS, and the peer-to-peer (P2P) model.
- FTP is a client-server protocol that enables data transfer over a network. An FTP server and an FTP client communicate with each other using TCP as the transport protocol.
- A distributed file system (DFS) is a fi le system that is distributed across several hosts. A DFS can provide hosts with direct access to the entire file system, while ensuring efficient management and data security.
- A name service, such as Domain Name System (DNS), and directory services such as Microsoft Active Directory, and Network Information Services (NIS), helps users identify and access a unique resource over the network.

- A peer-to-peer (P2P) file sharing model uses a peer-to-peer network. P2P enables client machines to directly share fi les with each other over a network.

7.4 Components of NAS

- NAS device has *two* key components (as shown in 7-3): **NAS head** and **storage**.
- In some NAS implementations, the storage could be external to the NAS device and shared with other hosts.
- NAS head includes the following components:
    - CPU and memory
    - One or more network interface cards (NICs), which provide connectivity to the client network.
    - An optimized operating system for managing the NAS functionality. It translates file-level requests into block-storage requests and further converts the data supplied at the block level to file data
    - NFS, CIFS, and other protocols for file sharing
    - Industry-standard storage protocols and ports to connect and manage physical disk resources

➢ The NAS environment includes clients accessing a NAS device over an IP network using file-sharing protocols.

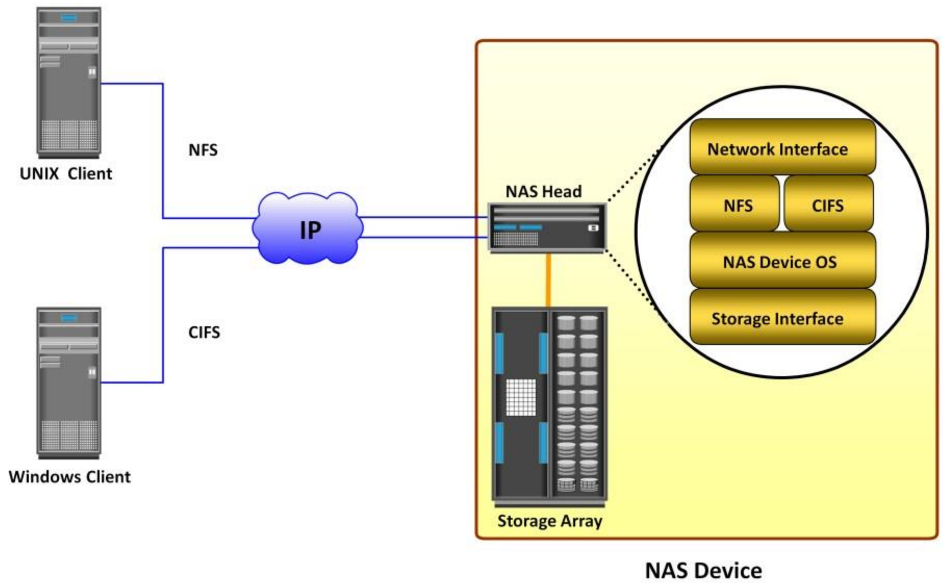

Fig 7-3: Components of NAS

7.5 NAS I/O Operation

➢ NAS provides *file-level data access* to its clients. File I/O is a high-level request that specifies the file to be accessed.

➢ Eg: a client may request a file by specifying its name, location, or other attributes. The NAS operating system keeps track of the location of files on the disk volume and converts client file I/O into block-level I/O to retrieve data.

➢ The process of handling I/Os in a NAS environment is as follows:

1. The requestor (client) packages an I/O request into TCP/IP and forwards it through the network stack. The NAS device receives this request from the network.
2. The NAS device converts the I/O request into an appropriate physical storage request,

which is a block-level I/O, and then performs the operation on the physical storage.

3. When the NAS device receives data from the storage, it processes and repackages thedata into an appropriate file protocol response.

4. The NAS device packages this response into TCP/IP again and forwards it to the client through the network.

- Fig 7-4 illustrates the NAS I/O operation

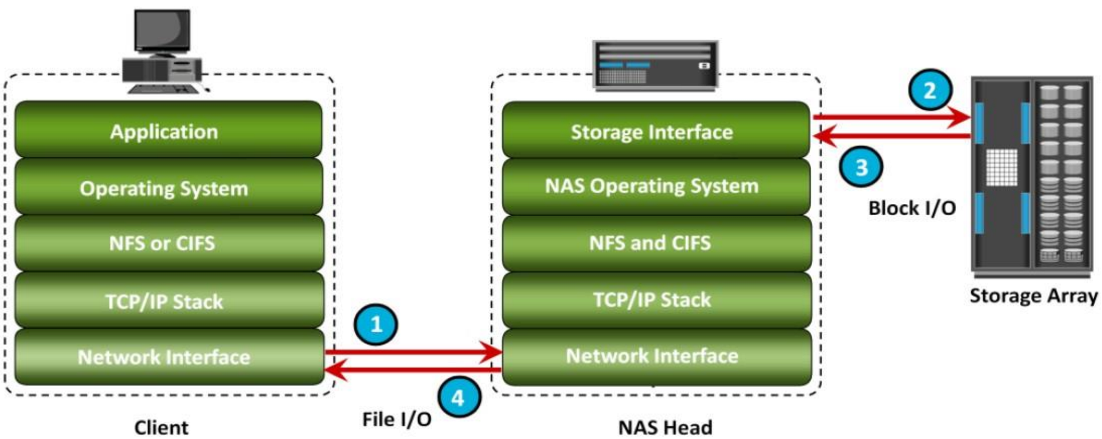

Fig 7-4: NAS I/O Operation

7.6NAS Implementations

- Three common NAS implementations are unified, gateway, and scale-out.
- The **unified NAS** consolidates NAS-based and SAN-based data access within a unified storage platform and provides a unified management interface for managing both the environments.
- In a **gateway** implementation, the NAS device uses external storage to store and retrieve data, and unlike unified storage, there are separate administrative tasks for the NAS device and storage.
- The **scale-out** NAS implementation pools multiple nodes together in a cluster. A node may consist of either the NAS head or storage or both. The cluster performs the NAS operation as a single entity.

7.6.1 Unified NAS

- Unified NAS performs file serving and storing of fi le data, along with providing access to block-level data.
- It supports both CIFS and NFS protocols for file access and iSCSI and FC protocols for block level access.
- A unified NAS contains one or more NAS heads and storage in a single system. NAS heads are connected to the storage controllers (SCs), which provide access to the storage.

7.6.2 Unified NAS Connectivity

- Each NAS head in a unified NAS has front-end Ethernet ports, which connect to the IP network. The front-end ports provide connectivity to the clients and service the file I/O requests.
- iSCSI and FC ports on a storage controller enable hosts to access the storage directly or through a storage network at the block level. Figure 7-5 illustrates an example of unified NAS connectivity.

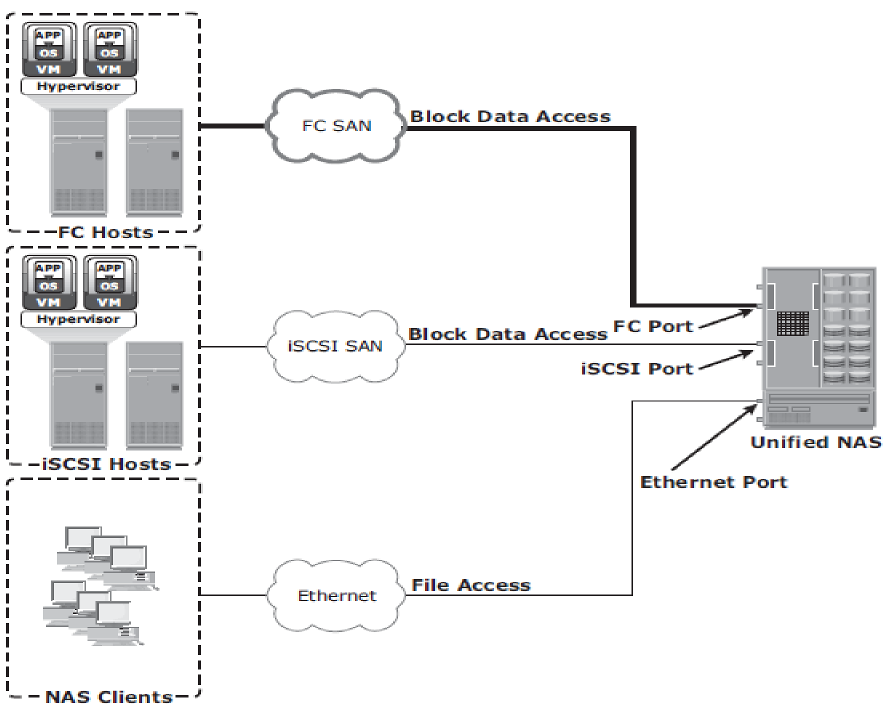

Figure 7-5: Unified NAS connectivity

7.6.3 Gateway NAS

- A gateway NAS device consists of one or more NAS heads and uses external and independently managed storage.
- Similar to unified NAS, the storage is shared with other applications that use block-level I/O.
- Management functions in this type of solution are more complex than those in a unified NAS environment because there are separate administrative tasks for the NAS head and the storage.
- The gateway NAS is more scalable compared to unified NAS because NAS heads and storage arrays can be independently scaled up when required.

7.6.4 Gateway NAS Connectivity

- In a gateway solution, the front-end connectivity is similar to that in a unified storage solution.
- Communication between the NAS gateway and the storage system in a gateway solution is achieved through a traditional FC SAN.
- To deploy a gateway NAS solution, factors, such as multiple paths for data, redundant fabrics, and load distribution, must be considered.
- Figure 7-6 illustrates an example of gateway NAS connectivity.
- Implementation of both unified and gateway solutions requires analysis of the SAN environment.

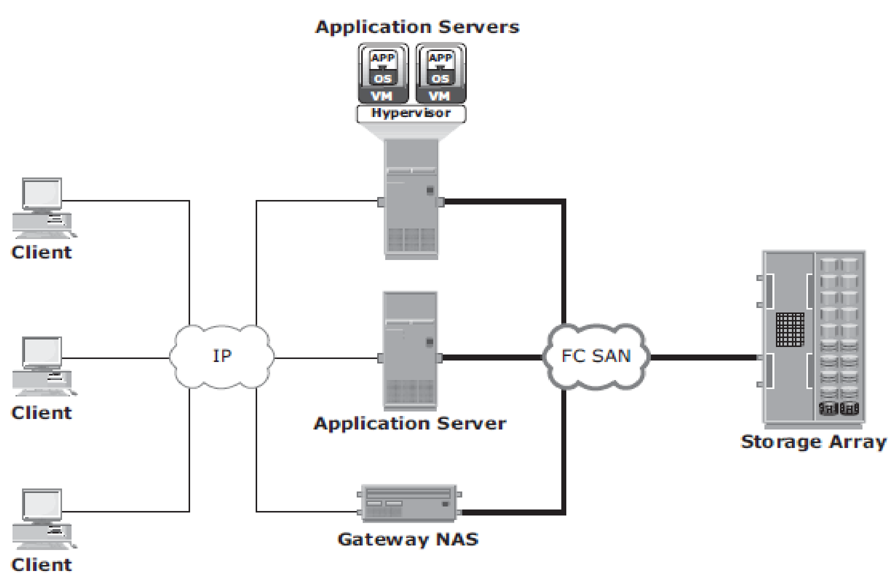

Figure 7-6: Gateway NAS connectivity

7.6.5 Scale-Out NAS

- Both unified and gateway NAS implementations provide the capability to scaleup their resources based on data growth and rise in performance requirements.
- Scaling up these NAS devices involves adding CPUs, memory, and storage to the NAS device. Scalability is limited by the capacity of the NAS device to house and use additional NAS heads and storage.
- Scale-out NAS enables grouping multiple nodes together to construct a clustered NAS system. A scale-out NAS provides the capability to scale its resources by simply adding nodes to a clustered NAS architecture.
- Scale-out NAS creates a single fi le system that runs on all nodes in the cluster. All information is shared among nodes, so the entire file system is accessible by clients connecting to any node in the cluster.
- Scale-out NAS is suitable to solve the "Big Data" challenges that enterprises and customers face today.

7.6.6 Scale-Out NAS Connectivity

- Scale-out NAS clusters use separate internal and external networks for back-end and front-end connectivity, respectively.
- An internal network provides connections for intracluster communication, and an external network connection enables clients to access and share file data.
- Each node in the cluster connects to the internal network. The internal network offers high throughput and low latency and uses high-speed networking technology, such as InfiniBand or Gigabit Ethernet.
- To enable clients to access a node, the node must be connected to the external Ethernet network. Redundant internal or external networks may be used for high availability
- Figure 7-7 illustrates an example of scale-out

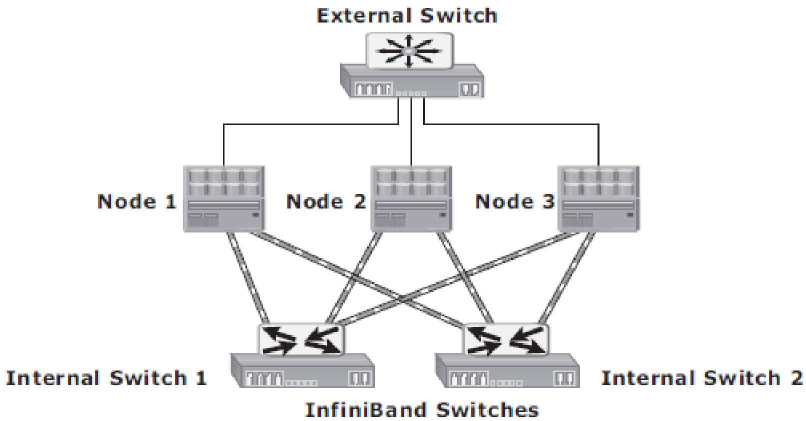

Figure 7-7: Scale-out NAS with dual internal and single external networks

7.7 NAS File Sharing Protocols

- NAS devices support multiple file-service protocols to handle file I/O requests
- Two common NAS file sharing protocols are:
  - Common Internet File System (CIFS)
  - Network File System (NFS)
- NAS devices enable users to share file data across different operating environments
- It provides a means for users to migrate transparently from one operating system to another

7.7.1 Network File System (NFS)

- NFS is a **client-server protocol** for file sharing that is commonly used on **UNIX systems**.
- NFS was originally based on the connectionless *User Datagram Protocol (UDP)*.
- It uses *Remote Procedure Call (RPC)* as a method of inter-process communication between two computers.
- The NFS protocol provides a set of RPCs to access a remote file system for the following operations:
    - Searching files and directories
    - Opening, reading, writing to, and closing a file
    - Changing file attributes
    - Modifying file links and directories
- NFS creates a connection between the client and the remote system to transfer data.
- NFSv3 and earlier is a stateless protocol
- It does not maintain any kind of table to store information about open files and associated pointers. Each call provides a full set of arguments - a file handle, a particular position to read or write, and the versions of NFS - to access files on the server .
- Currently, three versions of NFS are in use:

1. **NFS version 2 (NFSv2):** Uses *UDP* to provide a *stateless* network connection between a client and a server. Features, such as locking, are handled outside the protocol.
2. **NFS version 3 (NFSv3):** Uses *UDP or TCP*, and is based on the *stateless protocol* design. It includes some new features, such as a 64-bit file size, asynchronous writes, andadditional file attributes to reduce refetching.
3. **NFS version 4 (NFSv4):** Uses TCP and is based on a *stateful protocol* design. It offers enhanced security. The latest NFS version 4.1 is the enhancement of NFSv4 and includessome new features, such as session model, parallel NFS (pNFS), and data retention.

7.7.2 Common Internet File System (CIFS)

- CIFS is a *client-server application* protocol
- It enables clients to access files and services on remote computers over **TCP/IP**.
- It is a public, or open, variation of **Server Message Block (SMB)** protocol.
- It provides following features to ensure data integrity:
  - It uses file and record locking to prevent users from overwriting the work of another user on a file or a record.
  - It supports fault tolerance and can automatically restore connections and reopen files that were open prior to an interruption. This feature depends on whether an application is written to take advantage of this.
  - CIFS is a stateful protocol because the CIFS server maintains connection information regarding every connected client. If a network failure or CIFS server failure occurs, the client receives a disconnection notification. User disruption is minimized if the application has the embedded intelligence to restore the connection. However, if the embedded intelligence is missing, the user must take steps to reestablish the CIFS connection.
- Users refer to remote file systems with an easy-to-use file-naming scheme:
- Eg: \\server\share or \\servername.domain.suffix\share

7.8 Factors Affecting NAS Performance

NAS uses IP network; therefore, bandwidth and latency issues associated with IP affect NAS performance. Network congestion is one of the most significant sources of latency (Figure 7-8) in a NAS environment. Other factors that affect NAS performance at different levels follow:

1. **Number of hops**: A large number of hops can increase latency because IP processing is required at each hop, adding to the delay caused at the router.

2. **Authentication with a directory service such as Active Directory or NIS**: The authentication service must be available on the network with enough resources to accommodate

the authentication load. Otherwise, a large number of authentication requests can increase latency.

3. **Retransmission:** Link errors and buffer overflows can result in retransmission. This causes packets that have not reached the specified destination to be re-sent. Care must be taken to match both speed and duplex settings on the network devices and the NAS heads. Improper configuration might result in errors and retransmission, adding to latency.

4. **Overutilized routers and switches:** The amount of time that an overutilized device in a network takes to respond is always more than the response time of an optimally utilized or underutilized device. Network administrators can view utilization statistics to determine the optimum utilization of switches and routers in a network. Additional devices should be added if the current devices are overutilized

5. **File system lookup and metadata requests:** NAS clients access files on NAS devices. The processing required to reach the appropriate file or directory can cause delays. Sometimes a delay is caused by deep directory structures and can be resolved by flattening the directory structure. Poor file system layout and an overutilized disk system can also degrade performance.

6. **Over utilized NAS devices**: Clients accessing multiple fi les can cause high utilization levels on a NAS device, which can be determined by viewing utilization statistics. High memory, CPU, or disk subsystem utilization levels can be caused by a poor fi le system structure or insufficient resources in a storage subsystem.

7. **Over utilized clients**: The client accessing CIFS or NFS data might also be over utilized. An overutilized client requires a longer time to process the requests and responses. Specific performance-monitoring tools are available for various operating systems to help determine the utilization of client resources.

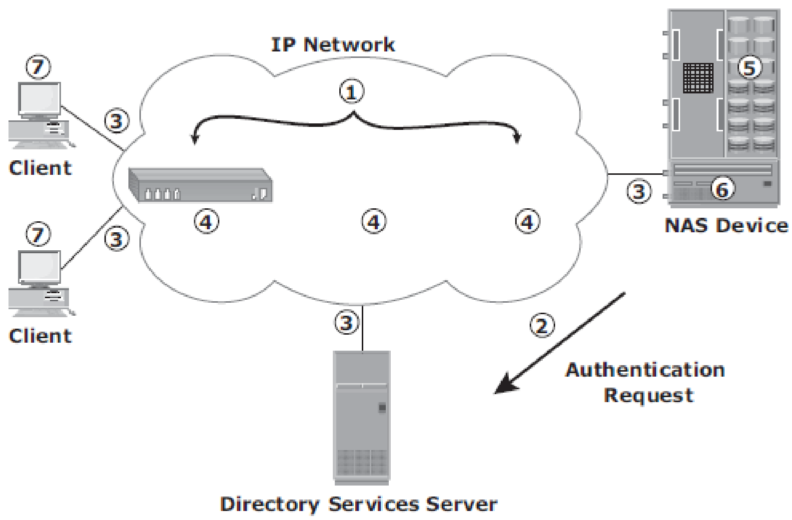

Figure 7-8: Causes of latency