Chapter 8
IP SAN

KEY CONCEPTS

iSCSI Protocol

Native and Bridged iSCSI

FCIP Protocol

Traditional SAN environments allow block I/O over Fibre Channel, whereas NAS environments allow file I/O over IP-based networks. Organizations need the performance and scalability of SAN plus the ease of use and lower TCO of NAS solutions. The emergence of IP technology that supports block I/O over IP has positioned IP for storage solutions.

IP offers easier management and better interoperability. When block I/O is run over IP, the existing network infrastructure can be leveraged, which is more economical than investing in new SAN hardware and software. Many long-distance, disaster recovery (DR) solutions are already leveraging IP-based networks. In addition, many robust and mature security options are now available for IP networks. With the advent of block storage technology that leverages IP networks (the result is often referred to as IP SAN), organizations can extend the geographical reach of their storage infrastructure.

IP SAN technologies can be used in a variety of situations. Figure 8-1 illustrates the co-existence of FC and IP storage technologies in an organization where mission-critical applications are serviced through FC, and business-critical applications and remote office applications make use of IP SAN. Disaster recovery solutions can also be implemented using both of these technologies.

Two primary protocols that leverage IP as the transport mechanism are iSCSI and Fibre Channel over IP (FCIP).

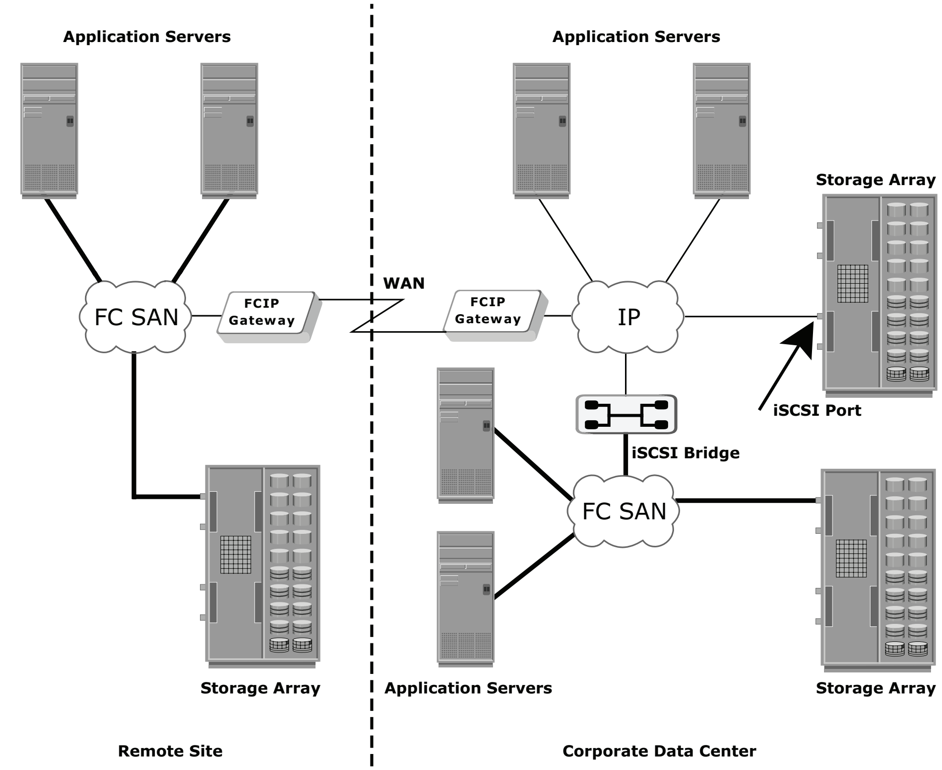


Figure 8-1: Co-existance of FC and IP storage technologies

iSCSI is the host-based encapsulation of SCSI I/O over IP using an Ethernet NIC card or an iSCSI HBA in the host. As illustrated in Figure 8-2 (a), IP traffic is routed over a network either to a gateway device that extracts the SCSI I/O from the IP packets or to an iSCSI storage array. The gateway can then send the SCSI I/O to an FC-based external storage array, whereas an iSCSI storage array can handle the extraction and I/O natively.

FCIP uses a pair of bridges (FCIP gateways) communicating over TCP/IP as the transport protocol. FCIP is used to extend FC networks over distances and/or an existing IP-based infrastructure, as illustrated in Figure 8-2 (b).

Today, iSCSI is widely adopted for connecting servers to storage because it is relatively inexpensive and easy to implement, especially in environments where an FC SAN does not exist. FCIP is extensively used in disaster-recovery implementations, where data is duplicated on disk or tape to an alternate site. This chapter describes iSCSI and FCIP protocols, components and topologies in detail.

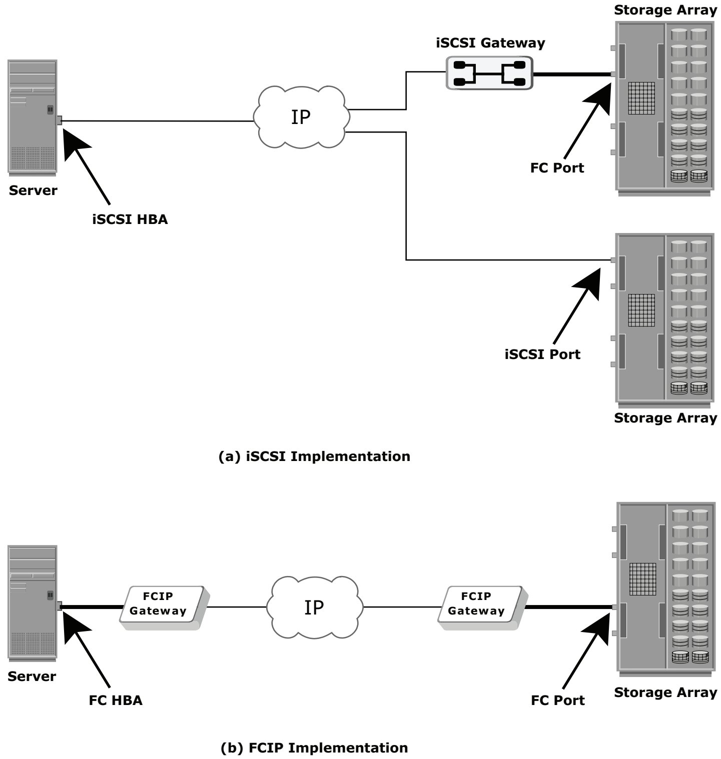


Figure 8-2: iSCSI and FCIP implementation

8.1 iSCSI

iSCSI is an IP-based protocol that establishes and manages connections between storage, hosts, and bridging devices over IP. iSCSI carries block-level data over IP-based networks, including Ethernet networks and the Internet. iSCSI is built on the SCSI protocol by encapsulating SCSI commands and data in order to allow these encapsulated commands and data blocks to be transported using TCP/IP packets.

8.1.1 Components of iSCSI

Host (initiators), targets, and an IP-based network are the principal iSCSI components. The simplest iSCSI implementation does not require any FC components. If an iSCSI-capable storage array is deployed, a host itself can act as an iSCSI initiator, and directly communicate with the storage over an IP network. However, in complex implementations that use an existing FC array for iSCSI connectivity, iSCSI gateways or routers are used to connect the existing FC SAN. These devices perform protocol translation from IP packets to FC packets and vice-versa, thereby bridging connectivity between the IP and FC environments.

8.1.2 iSCSI Host Connectivity

iSCSI host connectivity requires a hardware component, such as a NIC with a software component (iSCSI initiator) or an iSCSI HBA. In order to use the iSCSI protocol, a software initiator or a translator must be installed to route the SCSI commands to the TCP/IP stack.

A standard NIC, a TCP/IP offload engine (TOE) NIC card, and an iSCSI HBA are the three physical iSCSI connectivity options.

A standard NIC 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. However, the NIC provides no external processing power, which places additional overhead on the host CPU because it is required to perform all the TCP/IP and iSCSI processing.

If a standard NIC is used in heavy I/O load situations, the host CPU may become a bottleneck. *TOE NIC* help alleviate this burden. A TOE NIC offloads the TCP management functions from the host and leaves 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, requiring host CPU cycles.

An *iSCSI HBA* is capable of providing performance benefits, as it offloads the entire iSCSI and TCP/IP protocol stack from the host processor. Use of an iSCSI HBA is also the simplest way for implementing a boot from SAN environment via iSCSI. If there is no iSCSI HBA, modifications have to 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 very similar to the functionality of an FC HBA, but it is the most expensive option.

A fault-tolerant host connectivity solution can be implemented using host-based multipathing software (e.g., EMC PowerPath) regardless of the type of physical connectivity. Multiple NICs can also be combined via link aggregation technologies to provide failover or load balancing. Complex solutions may also include the use of vendor-specific storage-array software that enables the iSCSI host to connect to multiple ports on the array with multiple NICs or HBAs.

8.1.3 Topologies for iSCSI Connectivity

The topologies used to implement iSCSI can be categorized into two classes: native and bridged. *Native topologies* do not have any FC components; they perform all communication over IP. The initiators may be either directly attached to targets or connected using standard IP routers and switches. *Bridged topologies* enable the co-existence 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 SAN.

Native iSCSI Connectivity

If an iSCSI-enabled array is deployed, FC components are not needed for iSCSI connectivity in the native topology. In the example shown in Figure 8-3 (a), the array has one or more Ethernet NICs that are connected to a standard Ethernet switch and configured with an IP address and listening port. Once a client/initiator is configured with the appropriate target information, it connects to the array and requests a list of available LUNs. A single array port can service multiple hosts or initiators as long as the array can handle the amount of storage traffic that the hosts generate.

Many arrays provide more than one interface so that they can be configured in a highly available design or have multiple targets configured on the initiator. Some NAS devices are also capable of functioning as iSCSI targets, enabling file-level and block-level access to centralized storage. This offers additional storage options for environments with integrated NAS devices or environments that don't have an iSCSI/FC bridge.

Bridged iSCSI Connectivity

A bridged iSCSI implementation includes FC components in its configuration. Figure 8-3 (b) illustrates an existing FC storage array used to service hosts connected through iSCSI.

The array does not have any native iSCSI capabilities—that is, it does not have any Ethernet ports. Therefore, an external device, called a bridge, router, gateway, or a multi-protocol router, must be used to bridge the communication from the IP network to the FC SAN. These devices can be a stand-alone unit, or in many cases are integrated with an existing FC switch. In this configuration, the bridge device has Ethernet ports connected to the IP network, and FC ports connected to the storage. These ports are assigned IP addresses, similar to the ports on an iSCSI-enabled array.

The iSCSI initiator/host is configured with the bridge's IP address as its target destination. The bridge is also configured with an FC initiator or multiple initiators. These are called *virtual initiators* because there is no physical device, such as an HBA, to generate the initiator record.

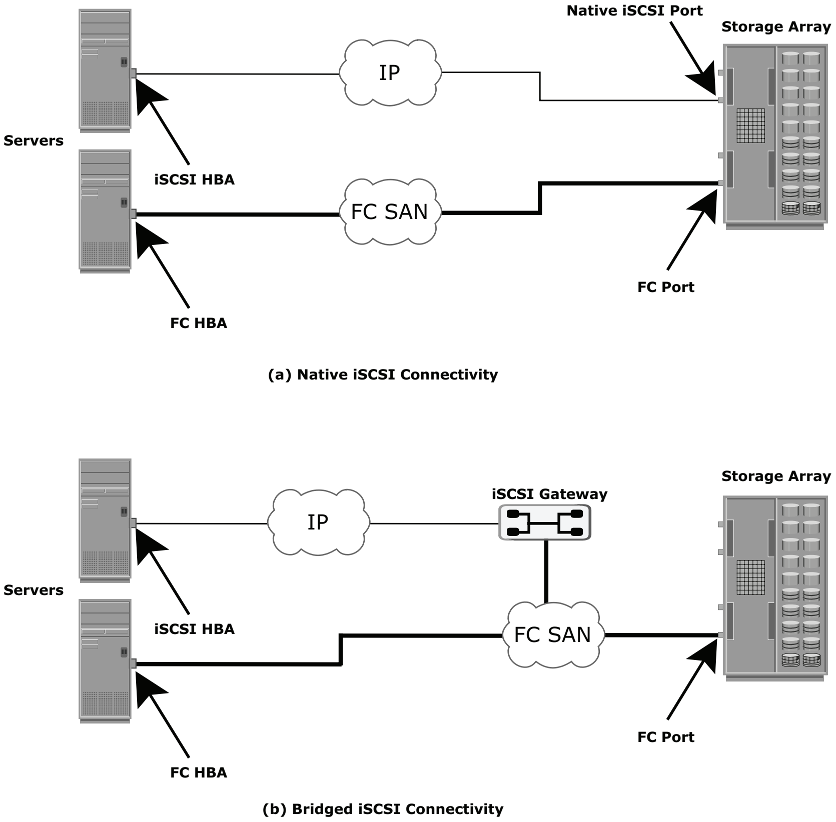


Figure 8-3: Native and bridged iSCSI connectivity

Combining FCP and Native iSCSI Connectivity

A combination topology can also be implemented. In this case, a storage array capable of connecting the FC and iSCSI hosts without the need for external bridging devices is needed (see Figure 8-3 [a]). These solutions reduce complexity, as they remove the need for configuring bridges. However, additional processing requirements are placed on the storage array because it has to accommodate the iSCSI traffic along with the standard FC traffic.

8.1.4 iSCSI Protocol Stack

The architecture of iSCSI is based on the client/server model. Figure 8-4 displays a model of the iSCSI protocol layers and depicts the encapsulation order of SCSI commands for their delivery through a physical carrier.

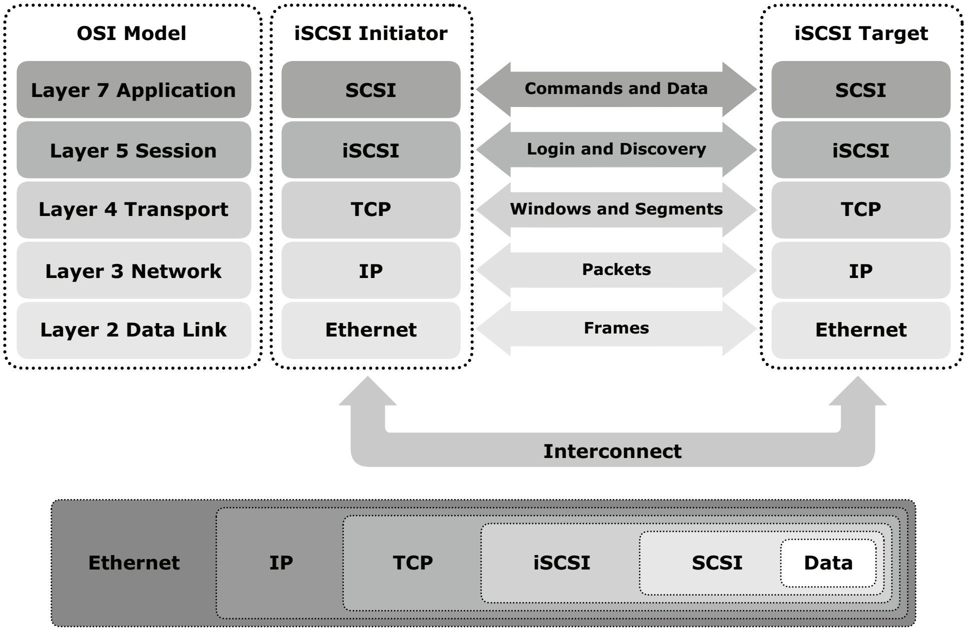


Figure 8-4: iSCSI protocol stack

SCSI is the command protocol that works at the application layer of the 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 initiators and targets.

iSCSI is the session-layer protocol that initiates a reliable session between a device that recognizes 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 service.

TCP is used to control 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 for each hop through a separate physical network.

Communication between an iSCSI initiator and target is detailed next.

8.1.5 iSCSI Discovery

An initiator must discover the location of the target on a 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* and *internet Storage Name Service (iSNS)*.

In SendTargets discovery, the initiator is manually configured with the target's network portal, which it uses to establish a discovery session with the iSCSI service on the target. The initiator issues the `SendTargets` command, and the target responds with the names and addresses of the targets available to the host.

iSNS (see Figure 8-5) enables the 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 know the targets that it can access, it can query the iSNS server for a list of available targets.

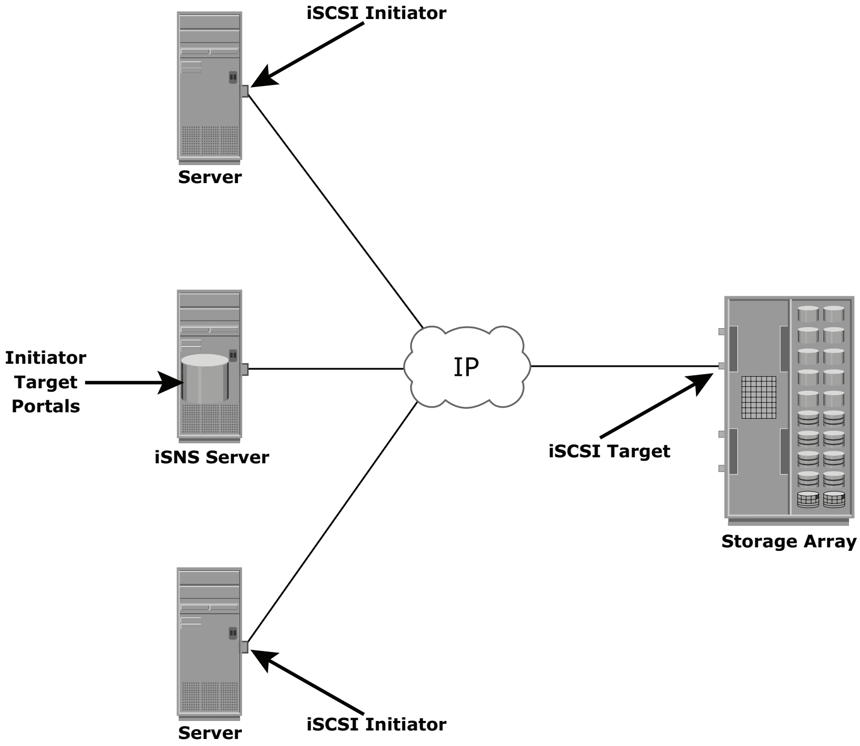


Figure 8-5: Discovery using iSNS

Discovery can also take place by using Service Location Protocol (SLP). However, this is less commonly used than `SendTargets` discovery and iSNS.

8.1.6 iSCSI Names

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

- **iSCSI Qualified Name (IQN):** An organization must own a registered domain name in order to generate iSCSI Qualified Names. This domain name does not have 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 transfer of domain names; the organization is required to have owned the domain name on that date. 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 of the storage device identifiers.

- **Extended Unique Identifier (EUI):** An EUI is a globally unique identifier based on the IEEE EUI-64 naming standard. An EUI comprises the eui prefix followed by a 16-character hexadecimal name, such as `eui.0300732A32598D26`.

  The 16-character part of the name includes 24 bits for the company name assigned by IEEE and 40 bits for a unique ID, such as a serial number. This allows for a more streamlined, although less user-friendly, name string because the resulting iSCSI name is simply eui followed by the hexadecimal WWN.

In either format, the allowed special characters are dots, dashes, and blank spaces. The iSCSI Qualified Name enables storage administrators to assign meaningful names to storage devices, and therefore manage those devices more easily.

Network Address Authority (NAA) is an additional iSCSI node name type to enable worldwide naming format as defined by the InterNational Committee for Information Technology Standards (INCITS) T11 - Fibre Channel (FC) protocols and used by Serial Attached SCSI (SAS). This format enables SCSI storage devices containing both iSCSI ports and SAS ports to use the same NAA-based SCSI device name. This format is defined by RFC3980, "T11 Network Address Authority (NAA) Naming Format for iSCSI Node Names."

8.1.7 iSCSI Session

An iSCSI session is established between an initiator and a target. A session ID (SSID), which includes an initiator ID (ISID) and a target ID (TSID), identifies a session. The session can be intended for one of the following:

- Discovery of available targets to the initiator and the location of a specific target on a network
- Normal operation of iSCSI (transferring data between initiators and targets)

TCP connections may be added and removed within a session. Each iSCSI connection within the session has a unique connection ID (CID).

8.1.8 iSCSI PDU

iSCSI initiators and targets communicate using iSCSI Protocol Data Units (PDUs). 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 Figure 8-6. The IP header provides packet-routing information that is used to move the packet across a network. The TCP header contains the information needed to guarantee the packet's delivery to the target. The iSCSI header describes how to extract SCSI commands and data for the target. iSCSI adds an optional CRC, known as the *digest*, beyond the TCP checksum and Ethernet CRC to ensure datagram integrity. The header and the data digests are optionally used in the PDU to validate integrity, data placement, and correct operation.

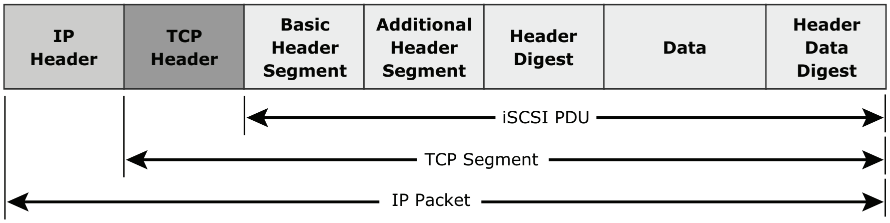


Figure 8-6: iSCSI PDU encapsulated in an IP packet

A message that is transmitted on a network is divided into a number of packets. If necessary, each packet can be sent by a different route across the network. Packets can arrive in a different order than the order in which they were sent. IP just delivers them. It's up to TCP to put them back in the right sequence. The target extracts the SCSI commands and data on the basis of information in the iSCSI header.

As shown in Figure 8-7, 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. Therefore, each IP packet and Ethernet frame can be used more efficiently because fewer packets and frames are required to transmit the SCSI information.

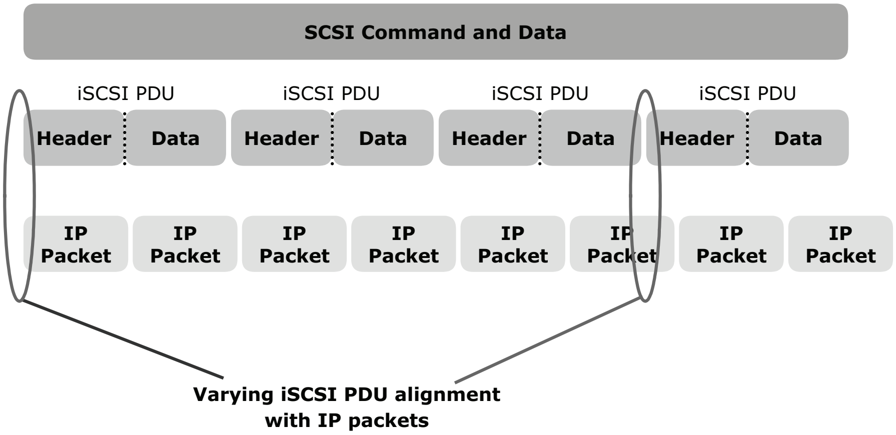


Figure 8-7: Alignment of iSCSI PDUs with IP packets

8.1.9 Ordering and Numbering

iSCSI communication between 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 to number all initiator-to-target command PDUs belonging to the session. This number is used to ensure 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. This ensures the correct order of data and commands at a target even when there are multiple TCP connections between an initiator and the target using portal groups.

Similar to command numbering, a *status sequence number (StatSN)* is used to sequentially number status responses, as shown in Figure 8-8. These unique numbers are established at the level of the TCP connection.

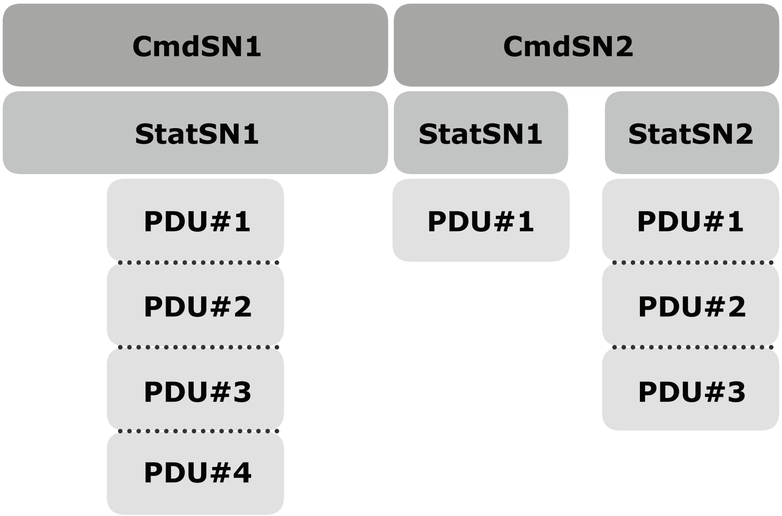


Figure 8-8: Command and status sequence number

A target sends the *request-to-transfer (R2T)* PDUs to the initiator when it is ready to accept data. *Data sequence number (DataSN)* is used to ensure in-order delivery of data within the same command. The DataSN and R2T sequence numbers are used to sequence data PDUs and R2Ts, respectively. Each of these sequence numbers is stored locally as an unsigned 32-bit integer counter defined by iSCSI. These numbers are communicated between the initiator and target in the appropriate iSCSI PDU fields during command, status, and data exchanges.

In the case of read operations, the DataSN begins at zero and is incremented by one for each subsequent data PDU in that command sequence. In the case of a write operation, the first unsolicited data PDU or the first data PDU in response to an R2T begins with a DataSN of zero and increments by one for each subsequent data PDU. R2TSN is set to zero at the initiation of the command and incremented by one for each subsequent R2T sent by the target for that command.

8.1.10 iSCSI Error Handling and Security

The iSCSI protocol addresses errors in IP data delivery. Command sequencing is used for flow control, the missing commands, and responses, and data blocks are detected using sequence numbers. Use of the optional digest improves communication integrity in addition to TCP checksum and Ethernet CRC.

The error detection and recovery in iSCSI can be classified into three levels: Level 0 = Session Recovery, Level 1 = Digest Failure Recovery and Level 2 = Connection Recovery. The error-recovery level is negotiated during login.

- **Level 0:** If an iSCSI session is damaged, all TCP connections need to be closed and all tasks and unfulfilled SCSI commands should be completed. Then, the session should be restarted via the repeated login.
- **Level 1:** Each node should be able to selectively recover a lost or damaged PDU within a session for recovery of data transfer. At this level, identification of an error and data recovery at the SCSI task level is performed, and an attempt to repeat the transfer of a lost or damaged PDU is made.
- **Level 2:** New TCP connections are opened to replace a failed connection. The new connection picks up where the old one failed.

iSCSI may be exposed to the security vulnerabilities of an unprotected IP network. Some of the security methods that can be used are IPSec and authentication solutions such as Kerberos and CHAP (challenge-handshake authentication protocol).

8.2 FCIP

Organizations are now looking for new ways to transport data throughout the enterprise, locally over the SAN as well as over longer distances, to ensure that data reaches all the users who need it. One of the best ways to achieve this goal is to interconnect geographically dispersed SANs through reliable, high-speed links. This approach involves transporting FC block data over the existing IP infrastructure used throughout the enterprise.

The FCIP standard has rapidly gained acceptance as a manageable, cost-effective way to blend the best of two worlds: FC block-data storage and the proven, widely deployed IP infrastructure. FCIP is a tunneling protocol that enables distributed FC SAN islands to be transparently interconnected over existing IP-based local, metropolitan, and wide-area networks. As a result, organizations now have a better way to protect, store, and move their data while leveraging investments in existing technology.

FCIP uses TCP/IP as its underlying protocol. In FCIP, the FC frames are encapsulated onto the IP payload, as shown in Figure 8-9. FCIP does not manipulate FC frames (translating FC IDs for transmission).

When SAN islands are connected using FCIP, each interconnection is called an *FCIP link*. A successful FCIP link between two SAN islands results in a fully merged FC fabric.

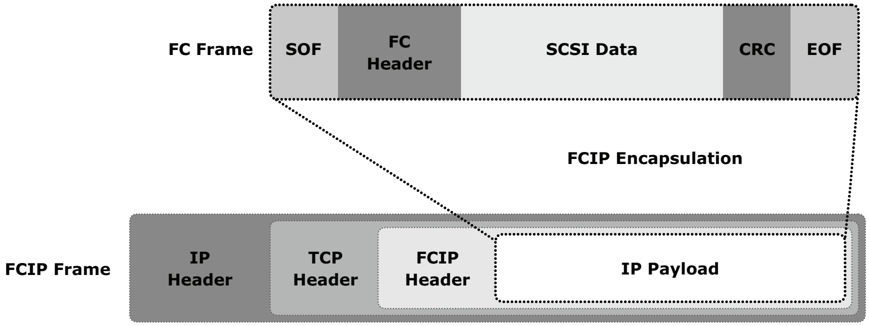


Figure 8-9: FCIP encapsulation

FCIP may require high network bandwidth when merging SANs or replicating or backing up data. FCIP does not handle data traffic throttling or flow control; these are controlled by the communicating FC switches and devices within the fabric.

8.2.1 FCIP Topology

An FCIP environment functions as if it is a single cohesive SAN environment. Before geographically dispersed SANs are merged, a fully functional layer 2 network exists on the SANs. This layer 2 network is a standard SAN fabric. These physically independent fabrics are merged into a single fabric with an IP link between them.

An FCIP gateway router is connected to each fabric via a standard FC connection (see Figure 8-10). The fabric treats these routers like layer 2 fabric switches. The other port on the router is connected to an IP network and an IP address is assigned to that port. This is similar to the method of assigning an IP address to an iSCSI port on a gateway. Once IP connectivity is established, the two independent fabrics are merged into a single fabric. When merging the two fabrics, all the switches and routers must have unique domain IDs, and the fabrics must contain unique zone set names. Failure to ensure these requirements will result in a segmented fabric. The FC addresses on each side of the link are exposed to the other side, and zoning or masking can be done to any entity in the new environment.

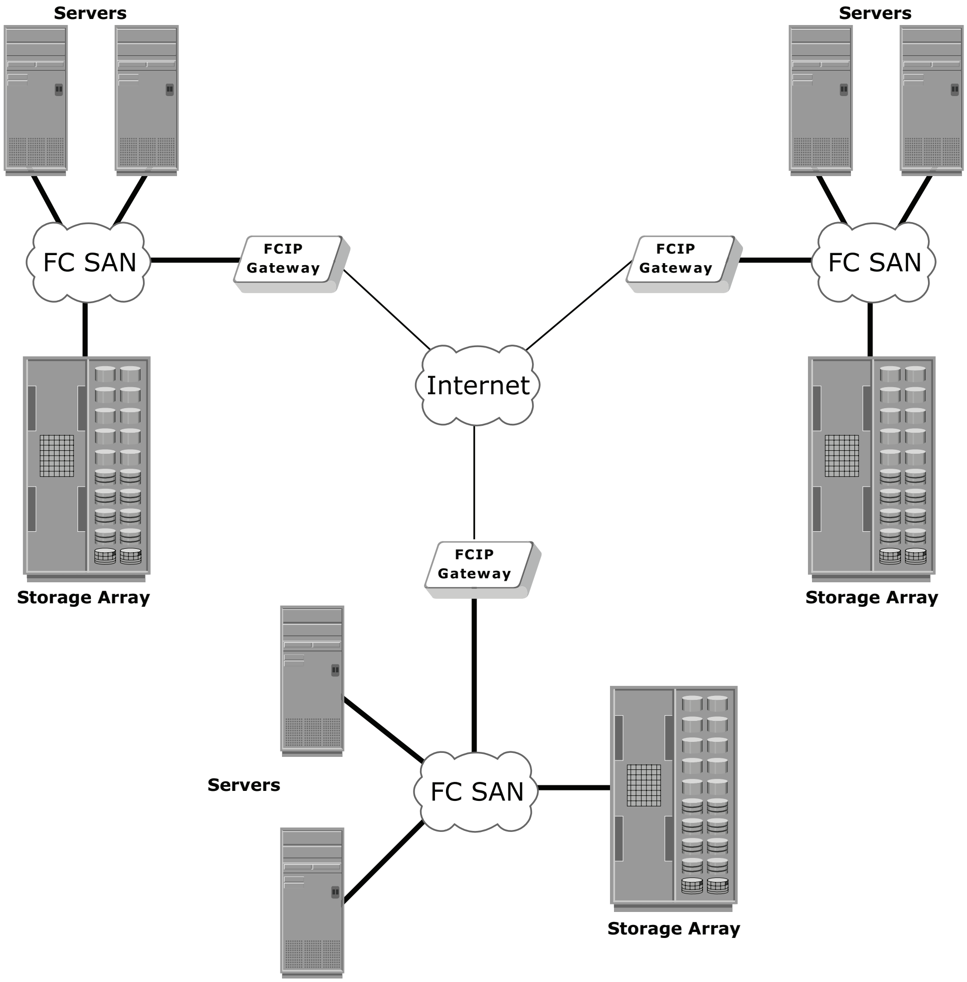


Figure 8-10: FCIP topology

8.2.2 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 consideration.

From the perspective of performance, multiple paths to multiple FCIP gateways from different switches in the layer 2 fabric eliminates single points of failure and provides increased bandwidth. In a scenario of extended distance, the IP network may be a bottleneck if sufficient bandwidth is not available. In addition, because FCIP creates a unified fabric, disruption in the underlying IP network can cause instabilities in the SAN environment. These include a segmented fabric, excessive RSCNs, and host timeouts.

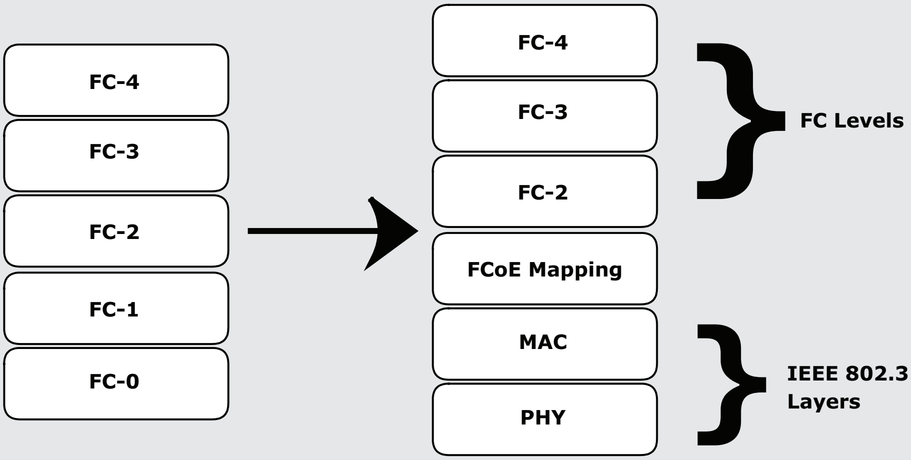
The vendors of FC switches have recognized some of the drawbacks related to FCIP and have implemented features to provide additional stability, such as the capability to segregate FCIP traffic into a separate virtual fabric.

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.

FIBRE CHANNEL OVER ETHERNET (FCOE)

FCoE is a mapping of FC frames over Gigabit Ethernet networks. Ethernet is used as the physical interface for carrying FC frames. Multi-function network/storage adapters are used for FC-to-Ethernet mapping.

FCoE maps FC natively over Ethernet while being independent of the Ethernet forwarding scheme, as shown in the following figure.



The FCoE protocol specification replaces the FC0 and FC1 layers of the FC stack with Ethernet. By retaining the native FC constructs, FCoE allows a seamless integration with existing FC networks and management software. For more information, visit `http://www.fcoe.com`**.**

Summary

iSCSI has enabled IT organizations to gain the benefits of storage networking architecture at reasonable costs. Storage networks can now be geographically distributed with the help of hybrid IP SAN technology, which enhances storage utilization across enterprises. FCIP has emerged as a solution for implementing viable business continuity across enterprises.

Because IP SANs are based on standard Ethernet protocols, the concepts, security mechanisms, and management tools are familiar to administrators. This has enabled the rapid adoption of IP SAN in organizations. The block-level I/O requirements of certain applications that cannot be made with NAS can be targeted for implementation with iSCSI.

This chapter detailed the two IP SAN technologies, iSCSI and FCIP. The next chapter focuses on CAS, another important storage networking technology that addresses the online storage and retrieval of content and long-term archives.

EXERCISES

1. **How does iSCSI handle the process of authentication? Research the available options.**
2. **List some of the data storage applications that could benefit from an IP SAN solution.**
4. **What are the major performance considerations for FCIP?**
5. **Research the multipathing software available for an iSCSI environment. Write a technical note on the features and functionality of EMC PowerPath support for iSCSI.**
6. **Research the iSCSI capabilities in a NAS device; provide use case examples.**
7. **A company is considering implementing storage. They do not have a current storage infrastructure to use, but they have a network that gives them good performance. Discuss whether native or bridged iSCSI should be used and explain your recommendation.**
8. **The IP bandwidth provided for FCIP connectivity seems to be constrained. Discuss its implications if the SANs that are merged are fairly large, with 500 ports on each side, and the SANs at both ends are constantly reconfigured.**
9. **Compared to a standard IP frame, what percentage of reduction can be realized in protocol overhead in an iSCSI configured to use jumbo frames with an MTU value of 9,000 bytes?**
10. **Why should an MTU value of at least 2,500 bytes be configured in a bridged iSCSI environment?**