CHAPTER 4

Overview of Modern SCSI Networking Protocols

The goal of this chapter is to quickly acclimate readers to the standard upper-layer storage protocols currently being deployed. To that end, we provide a conceptual description and brief procedural overview for each ULP. The procedural overviews are greatly simplified and should not be considered technically complete. Procedural details are provided in subsequent chapters.

iSCSI

This section provides a brief introduction to the Internet Small Computer System Interface (iSCSI) protocol.

iSCSI Functional Overview

As indicated in Chapter 2, "The OSI Reference Model Versus Other Network Models," iSCSI is a Small Computer System Interface (SCSI) Transport Protocol. The Internet Engineering Task Force (IETF) began working on iSCSI in 2000 and subsequently published the first iSCSI standard in 2004. iSCSI facilitates block-level initiator-target communication over TCP/IP networks. In doing so, iSCSI completes the storage over IP model, which supported only file-level protocols (such as Network File System [NFS], Common Internet File System [CIFS], and File Transfer Protocol [FTP]) in the past. To preserve backward compatibility with existing IP network infrastructure components and to accelerate adoption, iSCSI is designed to work with the existing TCP/IP architecture. iSCSI requires no special modifications to TCP or IP. All underlying network technologies supported by IP can be incorporated as part of an iSCSI network, but most early deployments are expected to be based solely on Ethernet. Other lower-layer technologies eventually will be leveraged as iSCSI deployments expand in scope. iSCSI is also designed to work with the existing SCSI architecture, so no special modifications to SCSI are required for iSCSI adoption. This ensures compatibility with a broad portfolio of host operating systems and applications.

iSCSI seamlessly fits into the traditional IP network model in which common network services are provided in utility style. Each of the IP network service protocols performs a single function very efficiently and is available for use by every "application" protocol.

iSCSI is an application protocol that relies on IP network service protocols for name resolution (Domain Name System [DNS]), security (IPsec), flow control (TCP windowing), service location (Service Location Protocol [SLP], and Internet Storage Name Service [iSNS]), and so forth. This simplifies iSCSI implementation for product vendors by eliminating the need to develop a solution to each network service requirement.

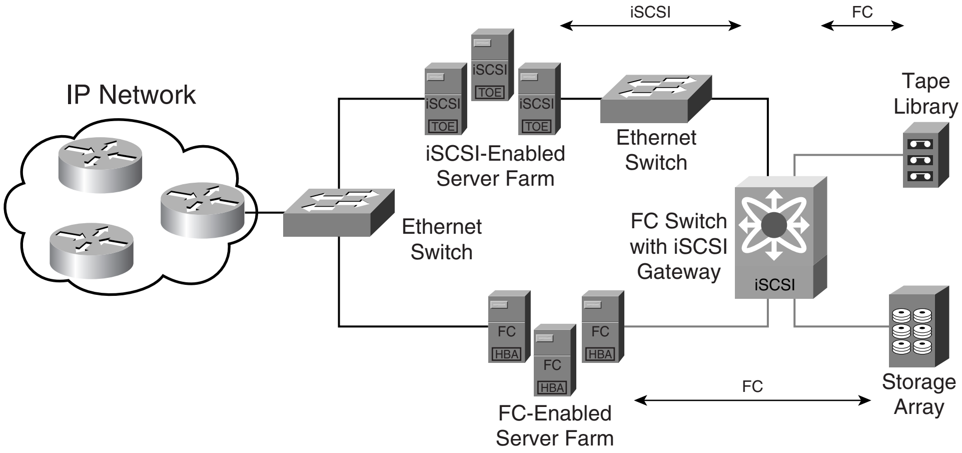
When the IETF first began developing the iSCSI protocol, concerns about the security of IP networks prompted the IETF to require IPsec support in every iSCSI product. This requirement was later deemed too burdensome considering the chip-level technology available at that time. So, the IETF made IPsec support optional in the final iSCSI standard (Request For Comments [RFC] 3720). IPsec is implemented at the OSI network layer and complements the authentication mechanisms implemented in iSCSI. If IPsec is used in an iSCSI deployment, it may be integrated into the iSCSI devices or provided via external devices (such as IP routers). The iSCSI standard stipulates which specific IPsec features must be supported if IPsec is integrated into the iSCSI devices. If IPsec is provided via external devices, the feature requirements are not specified. This allows shared external IPsec devices to be configured as needed to accommodate a wide variety of pass-through protocols. Most iSCSI deployments currently do not use IPsec.

One of the primary design goals of iSCSI is to match the performance (subject to underlying bandwidth) and functionality of existing SCSI Transport Protocols. As Chapter 3, “An Overview of Network Operating Principles,” discusses, the difference in underlying bandwidth of iSCSI over Gigabit Ethernet (GE) versus Fibre Channel Protocol (FCP) over 2-Gbps Fibre Channel (FC) is not as significant as many people believe. Another oft misunderstood fact is that very few 2-Gbps Fibre Channel Storage Area Networks (FC-SANs) are fully utilized. These factors allow companies to build block-level storage networks using a rich selection of mature IP/Ethernet infrastructure products at comparatively low prices without sacrificing performance. Unfortunately, many storage and switch vendors have propagated the myth that iSCSI can be used in only low-performance environments. Compounding this myth is the cost advantage of iSCSI, which enables cost-effective attachment of low-end servers to block-level storage networks. A low-end server often costs about the same as a pair of fully functional FC Host Bus Adapters (HBAs) required to provide redundant FC-SAN connectivity. Even with the recent introduction of limited-functionality HBAs, FC attachment of low-end servers is difficult to cost-justify in many cases. So, iSCSI is currently being adopted primarily for low-end servers that are not SAN-attached. As large companies seek to extend the benefits of centralized storage to low-end servers, they are considering iSCSI. Likewise, small businesses, which have historically avoided FC-SANs altogether due to cost and complexity, are beginning to deploy iSCSI networks.

That does not imply that iSCSI is simpler to deploy than FC, but many small businesses are willing to accept the complexity of iSCSI in light of the cost savings. It is believed that iSCSI (along with the other IP Storage [IPS] protocols) eventually can breathe new life into the Storage Service Provider (SSP) market. In the SSP market, iSCSI enables initiators secure access to centralized storage located at an SSP Internet Data Center (IDC) by removing the distance limitations of FC. Despite the current adoption trend in low-end environments, iSCSI is a very robust technology capable of supporting relatively high-performance applications. As existing iSCSI products mature and additional iSCSI products come to market, iSCSI adoption is likely to expand into high-performance environments.

Even though some storage array vendors already offer iSCSI-enabled products, most storage products do not currently support iSCSI. By contrast, iSCSI TCP Offload Engines (TOEs) and iSCSI drivers for host operating systems are widely available today. This has given rise to iSCSI gateway devices that convert iSCSI requests originating from hosts (initiators) to FCP requests that FC attached storage devices (targets) can understand. The current generation of iSCSI gateways is characterized by low port density devices designed to aggregate multiple iSCSI hosts. Thus, the iSCSI TOE market has suffered from low demand. As more storage array vendors introduce native iSCSI support in their products, use of iSCSI gateway devices will become less necessary. In the long term, it is likely that companies will deploy pure FC-SANs and pure iSCSI-based IP-SANs (see Figures 1-1 and 1-2, respectively) without iSCSI gateways, and that use of iSCSI TOEs will likely become commonplace. That said, iSCSI gateways that add value other than mere protocol conversion might remain a permanent component in the SANs of the future. Network-based storage virtualization is a good example of the types of features that could extend the useful life of iSCSI gateways. Figure 4-1 illustrates a hybrid SAN built with an iSCSI gateway integrated into an FC switch. This deployment approach is common today.

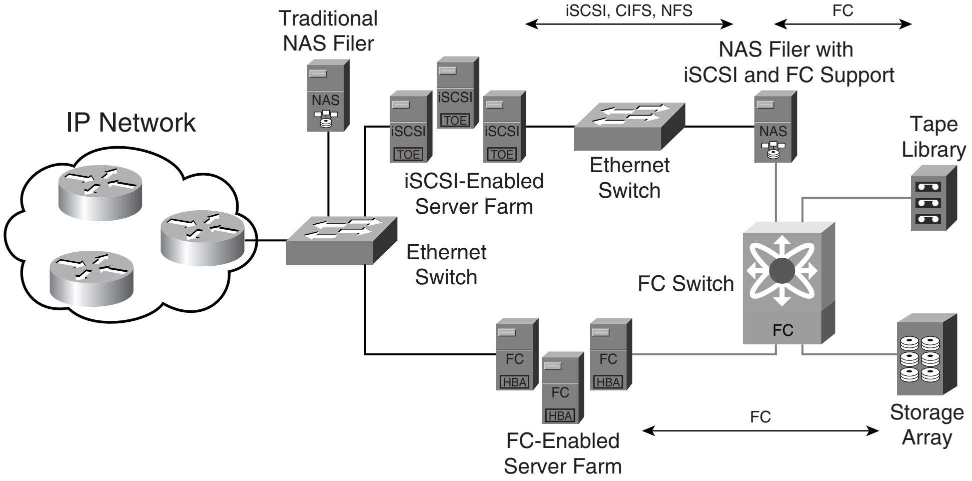
Figure 4-1 *Hybrid SAN Built with an iSCSI Gateway*



Another way to accomplish iSCSI-to-FCP protocol conversion is to incorporate iSCSI into the portfolio of protocols already supported by Network Attached Storage (NAS) filers. Because NAS filers natively operate on TCP/IP networks, iSCSI is a natural fit. Some NAS vendors already have introduced iSCSI support into their products, and it is expected that most (if not all) other NAS vendors eventually will follow suit. Another emerging trend in

NAS filer evolution is the ability to use FC on the backend. A NAS filer is essentially an optimized file server; therefore the problems associated with the DAS model apply equally to NAS filers and traditional servers. As NAS filers proliferate, the distributed storage that is captive to individual NAS filers becomes costly and difficult to manage. Support for FC on the backend allows NAS filers to leverage the FC-SAN infrastructure that many companies already have. For those companies that do not currently have an FC-SAN, iSCSI could be deployed as an alternative behind the NAS filers (subject to adoption of iSCSI by the storage array vendors). Either way, using a block-level protocol behind NAS filers enables very large-scale consolidation of NAS storage into block-level arrays. In the long term, it is conceivable that all NAS protocols and iSCSI could be supported natively by storage arrays, thus eliminating the need for an external NAS filer. Figure 4-2 illustrates the model in which an iSCSI-enabled NAS filer is attached to an FC-SAN on the backend.

Figure 4-2 *iSCSI-Enabled NAS Filer Attached to an FC-SAN*



iSCSI Procedural Overview

When a host attached to an Ethernet switch first comes online, it negotiates operating parameters with the switch. This is followed by IP initialization during which the host receives its IP address (if the network is configured for dynamic address assignment via Dynamic Host Configuration Protocol [DHCP]). Next, the host discovers iSCSI devices and targets via one of the methods discussed in Chapter 3, "An Overview of Network Operating Principles." The host then optionally establishes an IPsec connection followed by a TCP connection to each discovered iSCSI device. The discovery method determines what happens next.

If discovery is accomplished via manual or automated configuration, the host optionally authenticates each target within each iSCSI device and then opens a normal iSCSI session with each successfully authenticated target. SCSI Logical Unit Number (LUN) discovery is the final step. The semi-manual discovery method requires an additional intermediate step. All iSCSI sessions are classified as either discovery or normal. A discovery session is used exclusively for iSCSI target discovery. All other iSCSI tasks are accomplished using normal sessions. Semi-manual configuration requires the host to establish a discovery session with each iSCSI device. Target discovery is accomplished via the iSCSI **SendTargets** command. The host then optionally authenticates each target within each iSCSI device. Next, the host opens a normal iSCSI session with each successfully authenticated target and performs SCSI LUN discovery. It is common for the discovery session to remain open with each iSCSI device while normal sessions are open with each iSCSI target.

Each SCSI command is assigned an iSCSI Command Sequence Number (CmdSN). The iSCSI CmdSN has no influence on packet tracking within the SCSI Interconnect. All packets comprising SCSI commands, data, and status are tracked in flight via the TCP sequence-numbering mechanism. TCP sequence numbers are directional and represent an increasing byte count starting at the initial sequence number (ISN) specified during TCP connection establishment. The TCP sequence number is not reset with each new iSCSI CmdSN. There is no explicit mapping of iSCSI CmdSNs to TCP sequence numbers. iSCSI complements the TCP sequence-numbering scheme with PDU sequence numbers. All PDUs comprising SCSI commands, data, and status are tracked in flight via the iSCSI CmdSN, Data Sequence Number (DataSN), and Status Sequence Number (StatSN), respectively. This contrasts with the FCP model.

FCP

This section provides a brief introduction to FCP.

FCP Functional Overview

As indicated in Chapter 2, "The OSI Reference Model Versus Other Network Models," FCP is a SCSI Transport Protocol. FCP was produced by ANSI T10 to facilitate block-level initiator-target communication over FC networks. FCP development began in 1991, and the first standard was published in 1996. In the FC network model, FCP is an FC-4 protocol. FCP uses the existing frame formats and services defined by the FC specifications, so no special modifications to FC are required by FCP. In theory, this allows FCP to share an FC network with other ULPs, but the vast majority of FC networks are deployed exclusively for FCP. FCP works with the existing SCSI architecture, so no special modifications to SCSI are required for FCP adoption. This ensures compatibility with a broad portfolio of host operating systems and applications.

Like the IP network model, the FC network model provides a robust set of network services to "application" protocols (that is, FC-4 protocols). FCP leverages the network services defined by the FC specifications. This simplifies FCP implementation for product vendors by reducing development overhead. Note that even OSI Session Layer login procedures are defined by the FC specifications (not the FCP specifications). This contrasts with the IP network model, in which each "application" protocol specifies its own OSI session layer login procedures. That said, FC-4 protocols are not required to use the services defined by the FC specifications.

There is a general perception that FC networks are inherently secure because they are physically separate (cabled independently) from the Internet and corporate intranets. This tenet is erroneous, but the FC user community is not likely to realize their error until FC security breaches become commonplace. For example, the vast majority of hosts that are attached to an FC-SAN are also attached to an IP network. If a host is compromised via the IP network, the host becomes a springboard for the intruder to access the FC-SAN. Moreover, FC-SANs are commonly extended across IP networks for disaster recovery applications. Such SAN extensions expose FC-SANs to a wide variety of attacks commonly perpetrated on IP networks. Authentication and encryption mechanisms are defined in the FC specifications, so FCP does not define its own security mechanisms. Unlike iSCSI, no security mechanisms are mandatory for FCP deployment. This fact and the misperception about the nature of FC network security have resulted in the vast majority of FC-SANs being deployed without any authentication or encryption.

Because FCP can be transported only by FC, the adoption rate of FCP is bound to the adoption rate of FC. The adoption rate of FC was relatively low in the late 1990s, in part because of the comparatively high cost of FC infrastructure components, which relegated FCP to high-end servers hosting mission-critical applications. Around 2000, FC adoption reached critical mass in the high-end server market. Simultaneously, performance improvements were being realized as companies began to view switched FC networks as the best practice instead of Fibre Channel Arbitrated Loop (FC-AL). In the years following, the adoption rate of switched FC increased dramatically. Consequently, FC prices began to drop and still are dropping as the FC market expands and competition increases. Furthermore, in response to competition from iSCSI, some FC HBA vendors have recently introduced "light" versions of their HBAs that provide less functionality at a lower cost than traditional FC HBAs. FCP is now being used by mid-range and high-end servers hosting a wide variety of business applications.

In the traditional FC-SAN design, each host and storage device is dual-attached to the network. (As previously noted, there are some exceptions to this guideline.) This is primarily motivated by a desire to achieve 99.999 percent availability. Conventional wisdom suggests that the network should be built with redundant switches that are not interconnected to achieve 99.999 percent availability. In other words, the network is actually two separate networks (commonly called path A and path B), and each end node (host or storage) is connected to both networks. Some companies take the same approach with their traditional IP/Ethernet networks, but most do not for reasons of cost. Because the traditional FC-SAN design doubles the cost of network implementation, many companies are actively seeking alternatives. Some companies are looking to iSCSI as the answer, and others are considering single path FC-SANs. Figures 3-12 and 3-13 illustrate typical dual path FC-SAN designs.

FC switches are market-classified as director class or fabric class. A director-class FC switch has no single point of failure in its architecture and provides 99.999 percent availability. Director-class FC switches also tend to have very high port density. By contrast, a fabric-class FC switch is similar to a traditional Ethernet switch in that it does not provide 99.999 percent availability. Fabric-class FC switches also tend to have limited port density. Some FC switch vendors employ completely different architectures in their director-class products versus their fabric-class products. This can result in functional disparity. In large-scale deployments, port density requirements often dictate the use of director-class FC switches. However, smaller environments can supplant the use of dual fabric-class FC switches with the use of a single director-class FC switch. This approach appeals to some companies because it fully protects the switch investment as the company grows, and it allows the company to take advantage of director-class functionality that might be missing from the fabric-class switches. However, availability can suffer if the director-class FC switch is physically damaged, or if a software bug in the switch's operating system causes an unplanned outage. Only dual path FC-SANs can protect against these risks.

A key difference between the FC and IP network models is support for routing protocols in the end nodes. Nearly all hosts attached to an IP network implement an IP routing protocol or static IP routing capability. If more than one router is attached to an Ethernet network, each host can decide which router to use when forwarding packets to non-local subnets. However, the forwarding decision is made at the network layer, not at the data-link layer. FC attached hosts do not implement a network layer protocol. Thus, an FC attached host merely forwards frames to the attached FC switch. This is equivalent to Ethernet frame processing in an IP attached host. Note that FC switches can perform load balancing, just as Ethernet switches and IP routers can. Chapter 10, "Routing and Switching Protocols," provides more detail about frame/packet forwarding mechanisms within networks. Chapter 11, "Load Balancing," provides more detail about load-balancing mechanisms within networks and hosts.

FCP Procedural Overview

When a host attached to an FC switch first comes online, it logs into the switch to exchange operating parameters and receive its FC address. Next, the host establishes an FC connection to the Fibre Channel Name Server (FCNS) and discovers FCP targets as discussed in Chapter 3, "An Overview of Network Operating Principles." The host then establishes an FC connection to each discovered FC device that contains a target. An FCP session is then established with each target. LUN discovery follows.

Each SCSI command is mapped to an FCP I/O operation. Each FCP I/O operation is identified by its Fully Qualified Exchange Identifier (FQXID), which is composed of the initiator's FC address, the target's FC address, an FC Exchange Identifier assigned by the initiator, and an FC Exchange Identifier assigned by the target. An FC Exchange Identifier is integral to frame tracking within the SCSI Interconnect. All frames comprising SCSI commands, data, and status are tracked in flight via the FC Sequence ID (SEQ_ID) and

Sequence Count (SEQ_CNT) mechanisms. Each SEQ_ID value identifies a single sequence of related frames within the context of an Exchange Identifier. Each SEQ_CNT value identifies a single frame within the context of a SEQ_ID. SEQ_ID values do not have to increase contiguously from one Exchange to the next. This contrasts with the iSCSI model.

FCIP

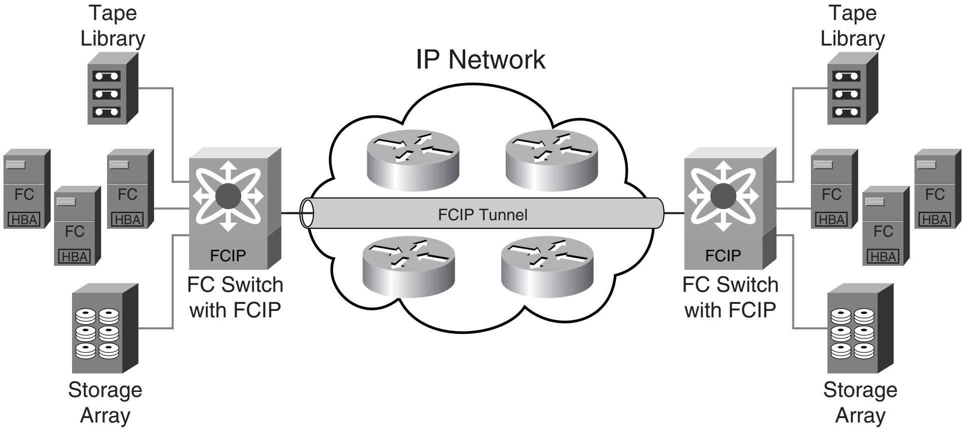
This section provides a brief introduction to Fibre Channel Over TCP/IP (FCIP).

FCIP Functional Overview

As indicated in Chapter 2, "The OSI Reference Model Versus Other Network Models," FCIP maps to the OSI session layer. The IETF began working on FCIP in 2000 and published the first standard in 2004 (RFC 3821). FCIP provides FC backbone functionality as defined in the ANSI T11 FC-BB series of specifications. Like iSCSI, FCIP seamlessly fits into the traditional IP network service model. This simplifies FCIP implementation for product vendors by reducing development overhead. IPsec support is mandatory for every FCIP product, and the FCIP standard stipulates which specific IPsec features must be supported. That said, use of IPsec is optional, and most FCIP deployments currently do not use IPsec.

FCIP enables interconnection of two switched FC-SANs using TCP/IP. This enables "distance applications" such as data replication from a primary site to a disaster recovery site. A point-to-point tunnel is created through an IP network, and FC traffic is transparently encapsulated or de-encapsulated at the tunnel endpoints. For multi-site connectivity, a separate tunnel must be created between each pair of FC-SANs. For this reason, each FCIP entity is architecturally capable of supporting multiple tunnels simultaneously. Each FCIP entity acts like a host on the IP network and does not require support for IP routing protocols. FCIP tunnels are long-lived. Once a tunnel is established, the two FC-SANs merge to form a single logical FC-SAN. All FC switches on each side of the tunnel see the remote FC switches as if they were local. All FC devices at both ends of the tunnel share a single address space. The FCIP tunnel is completely transparent to all FC devices. So, all aspects of the FC network architecture operate as if the tunnel were not present, and FC timers must be enforced end-to-end across the tunnel. FCIP's transparency enables all FC-4 protocols to be transported between the connected FC-SANs. FC-ALs can be attached to the FC switches within each FC-SAN, but low-level FC-AL signals (called primitives) cannot traverse an FCIP tunnel. This is not a problem because FCIP entities are either integrated into an FC switch or embodied in an external bridge device that connects to an FC switch, so low-level FC-AL signals never reach the FCIP entities. Figure 4-3 illustrates an FCIP tunnel connecting two physical FC-SANs.

Figure 4-3 *FCIP Tunnel Connecting Two Physical FC-SANs*



The two switches at the FCIP tunnel endpoints establish a standard FC inter-switch link (ISL) through the tunnel. Essentially, the FCIP tunnel appears to the switches as a cable. Each FCIP tunnel is created using one or more TCP connections. Multiple tunnels can be established between a pair of FC-SANs to increase fault tolerance and performance. Each tunnel carries a single FC ISL. Load balancing across multiple tunnels is accomplished via FC mechanisms just as would be done across multiple FC ISLs in the absence of FCIP. As mentioned in Chapter 3, “An Overview of Network Operating Principles,” encapsulation is accomplished per the Fibre Channel Frame Encapsulation (FC-FE) specification (RFC 3643). Eight bytes of the encapsulation header are used by each encapsulating protocol (such as FCIP) to implement protocol-specific functionality. The remainder of the encapsulation header is used for purposes common to all encapsulating protocols, such as identifying the encapsulating protocol and enforcing FC timers end-to-end.

Connectivity failures within the transit IP network can disrupt FC operations. Obviously, a circuit failure that results in FCIP tunnel failure will segment the FC-SAN and prevent communication between the FC-SAN segments. Unfortunately, the effect of the disruption is not limited to cross-tunnel traffic. Local connectivity is temporarily disrupted in each FC-SAN segment while the FC routing protocol reconverges and Registered State Change Notifications (RSCNs) are processed by end nodes. Additionally, one of the FC-SAN segments must select a new principle switch (see Chapter 5, “The OSI Physical and Data-Link Layers,” for details). The local effect of routing protocol reconvergence and principal switch selection can be eliminated via proprietary isolation techniques, but there currently is no mechanism within the FCIP standard to isolate FC-SANs from IP connectivity failures. This is generally considered to be the only significant drawback of FCIP.

FCIP Procedural Overview

Following data-link layer initialization, IP initialization occurs. The FCIP tunnel parameters can be configured manually or discovered via SLPv2. Once the tunnel parameters are known, an IPsec connection is optionally established followed by TCP connection establishment. The FCIP endpoint that initiated the TCP connection (the tunnel initiator) then transmits an FCIP Special Frame (FSF). The FSF contains the FC identifier and FCIP endpoint identifier of the tunnel initiator, the FC identifier of the intended destination, and a 64-bit randomly selected number that uniquely identifies the FSF. The receiver verifies that the contents of the FSF match its local configuration. If the FSF contents are acceptable, the unmodified FSF is echoed back to the tunnel initiator. After the tunnel initiator receives and verifies the FSF, the FCIP tunnel may carry FC traffic.

NOTE The term *identifier* is used generically in this section. As discussed in Chapter 5, "The OSI Physical and Data-Link Layers," the terms *identifier* and *name* each have specific meaning throughout the subsequent chapters of this book.

A time stamp is inserted into the header of each FCIP packet transmitted. The receiver checks the time stamp in each packet. If the time stamp indicates that the packet has been in flight longer than allowed by the FC timers, the packet is dropped. TCP is not responsible for retransmitting the dropped packet because the packet is dropped after TCP processing completes. FCP and FC error detection and recovery mechanisms are responsible for retransmitting the lost FC frame or notifying SCSI.

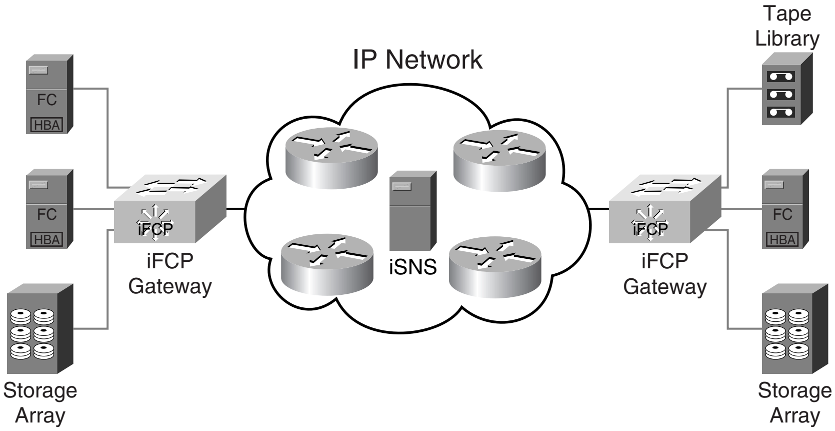
iFCP

This section provides a brief introduction to Internet Fibre Channel Protocol (iFCP).

iFCP Functional Overview

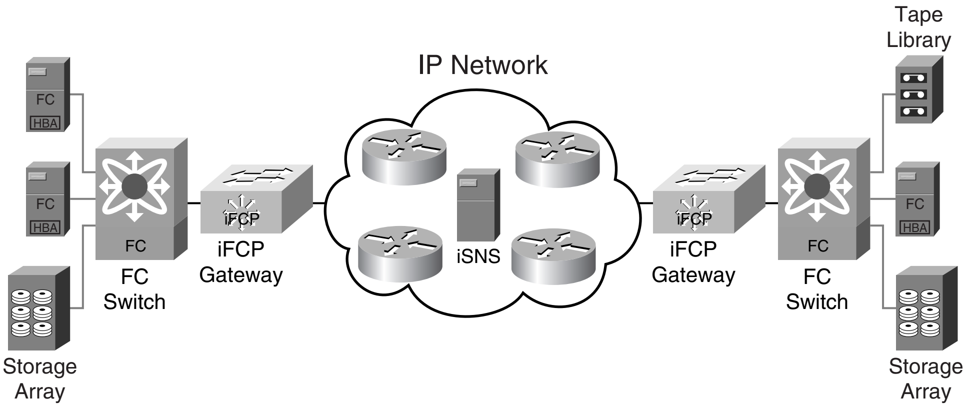
As indicated in Chapter 2, "The OSI Reference Model Versus Other Network Models," iFCP maps to the OSI session layer. The IETF began working on iFCP in 2000 and published the first standard in 2005 (RFC 4172). iFCP originally sought to supplant FC switching and routing mechanisms with Ethernet switching and IP routing mechanisms by replacing the FC fabric with an IP network. However, end nodes would be FC-attached to iFCP gateway devices. Only the iFCP gateway devices would communicate with each other via TCP/IP. The login procedures of FC would still be used by end nodes, but the IP network would provide the FC network services (in particular, the FCNS and FC Zone Server [FCZS]). Rather than augment the existing IP network services (such as SLP), a new service (iSNS) was created to provide the functionality of the FCNS and FCZS. As iSCSI development progressed in parallel to iFCP, modifications were made to iSNS to meet the needs of iSCSI. However, iSCSI can be deployed without iSNS. By contrast, iFCP requires iSNS. Figure 4-4 illustrates the original design concept of iFCP that was submitted to the IETF.

Figure 4-4 *Original iFCP Design Concept*



The motivation behind this approach is to reduce the total solution cost by leveraging cost-effective IP-based technology and widely available IP skill sets, extend the reach of FC attached devices beyond the FC limits, and enable the integration of FC and IP management operations. Unfortunately, the cost savings are undermined by the requirement for end nodes to be attached via FC. If the end nodes were attached via IP/Ethernet, the cost would be lower, but the solution would closely resemble iSCSI. Because iSCSI was designed to provide an IP/Ethernet alternative to FC-SANs, iSCSI provides a much more elegant and cost-effective solution than iFCP. Another challenge for iFCP is that only one vendor produces iFCP gateways today, and its iFCP products do not currently provide sufficient FC port densities to accommodate the connectivity requirements of most modern FC-SANs. So, iFCP gateways are usually deployed in conjunction with FC switches. Combined, these factors relegate iFCP usage to FC-SAN interconnectivity. Thus, iFCP competes against FCIP despite the original iFCP design goals. Figure 4-5 illustrates the current deployment practice.

Figure 4-5 *Current iFCP Deployment Practice*



The remainder of this section focuses on FC-SAN interconnectivity. iFCP gateways can operate in address transparency mode so that all FC devices share a single address space across all connected FC-SANs. This mode allows IP network failures to disrupt the attached FC-SANs just as FCIP does. For this reason, iFCP is rarely deployed in address transparency mode, and iFCP gateway support for address transparency mode is optional. iFCP gateways can also operate in address-translation mode. Devices in each FC-SAN communicate with devices in other FC-SANs using FC addresses allocated from the local FC-SAN address space. In this mode, the effect of IP network failures is mitigated. Each FC-SAN operates autonomously, as does the IP network. Network services are provided to FC attached devices via the FC switches. The state of FC network services in each FC-SAN must be replicated to the iSNS for propagation to other FC-SANs. Support for address translation mode is mandatory. Translation mode is customary in almost every deployment, so the remainder of this section focuses on translation mode.

iFCP operation is transparent to end nodes, and encapsulation is accomplished per the FC-FE specification (RFC 3643). However, connectivity across the IP network is handled differently than FCIP. Instead of creating a single tunnel to carry all FC traffic, each iFCP gateway creates a unique iFCP session to the appropriate destination iFCP gateway for each initiator-target pair that needs to communicate. This model might work well in the originally intended iFCP deployment scenario, but it can impede performance in the current iFCP deployment scenario by limiting the size of the TCP window available to each iFCP session. Two factors complicate this potential problem. First, iFCP sessions are created dynamically in response to PLOGI requests and are gracefully terminated only in response to LOGO requests. Second, PLOGI sessions are typically long-lived.

Like FCIP, IPsec support is mandatory for every iFCP product, and the iFCP standard stipulates which specific IPsec features must be supported. That said, use of IPsec is optional, and most iFCP deployments currently do not use IPsec. iFCP supports attachment of FC-ALs to FC switches within each FC-SAN, but low-level FC-AL signals (primitives) cannot traverse an iFCP session. This is not a problem because each iFCP gateway is usually connected to an FC switch, so FC-AL primitives never enter the iFCP gateways. iFCP gateways act like hosts on the IP network and do not require support for IP routing protocols. Multiple iFCP gateways may be deployed in each FC-SAN to increase fault tolerance and performance. Load balancing iFCP sessions across multiple iFCP gateways is implementation-specific. iFCP supports all FC-4 protocols when operating in transparent mode. However, it is possible for address translation mode to prevent certain FC-4 protocols from operating properly. Currently, iFCP is deployed only in FCP environments.

iFCP Procedural Overview

Following data-link layer initialization, IP initialization occurs. Next, iFCP gateways discover each other via iSNS. Likewise, configuration parameters for the iFCP fabric are discovered via iSNS. Once the iFCP fabric parameters are known, an IPsec connection is optionally established between each pair of iFCP gateways. As FC devices register in the

FCNS of each FC-SAN, the attached iFCP gateway propagates the registration information to the iSNS. The iSNS then propagates the information to each of the other iFCP gateways. Upon receipt, each iFCP gateway updates the FCNS of its attached FC-SAN with the remote node information and creates an entry in its address translation table for the remote node. At this point, the iFCP fabric is ready for initiator-target communication. TCP connections can be handled in two ways. An iFCP gateway can proactively establish and maintain multiple TCP connections to other gateways. These are called unbound TCP connections. When a PLOGI request is received, the iFCP gateway creates an iFCP session and binds it to one of the unbound TCP connections. Alternately, an iFCP gateway can wait until it receives a PLOGI request and then establish a TCP connection immediately followed by iFCP session establishment. While a TCP connection is bound to an iFCP session, it cannot be used by other iFCP sessions. iFCP enforces FC frame lifetimes in the same manner as FCIP. Likewise, detection and recovery of FC frames that are lost due to timeout are handled by FCP and FC. Due to limited vendor support and a low adoption rate, further examination of iFCP is currently outside the scope of this book.

Summary

This chapter provides a high-level overview of the upper-layer network protocols that comprise the SAM Transport Protocols. The protocols reviewed are all standards and, with the exception of iFCP, are commonly deployed in modern storage networks. The use of iSCSI and FCP for initiator-target communication is discussed, as is the use of FCIP and iFCP for long-distance FC-SAN connectivity across IP networks. The overviews in this chapter complement the information provided in Chapter 3, "An Overview of Network Operating Principles," in an effort to prime readers for the technical details provided in Part II, "The OSI Layers." Part II begins by examining the details of operation at OSI Layers 1 and 2.

Review Questions

1 How does iSCSI complement the traditional storage over IP model?

2 Is iSCSI capable of supporting high-performance applications?

3 How does the FC network service model resemble the IP network service model?

4 What is the guiding principle in traditional FC network designs?

5 What does FCIP create between FC-SANs to facilitate communication?

6 Which FC-4 protocols does FCIP support?

7 Is iFCP currently being deployed for its originally intended purpose?

8 Which address mode is most commonly used in modern iFCP deployments?