Fibre Channel over Internet Protocol Basics for Mainframers

By Steve Guendert, Ph.D.

Over the past decade, extension networks for storage have become commonplace and continue to grow in size and importance. Growth isn't limited to new deployments but also involves the expansion of existing deployments. Requirements for data protection will never ease, as the economies of many countries depend on successful and continued business operations; thus, laws have been passed mandating data protection. Modern-day dependence on remote data replication (RDR) means there's little tolerance for lapses that leave data vulnerable to loss. In IBM mainframe environments, reliable and resilient networks—to the point of no frame loss and in-order frame delivery—are necessary for error-free operation, high performance and operational ease. This improves availability, reduces risk and operating expenses and, most important of all, reduces risk of data loss.

A previous article, "Storage Networking Business Continuity Solutions" (*Enterprise Tech Journal*, October/November 2013, available at http://entsys.me/ixond), introduced the various topologies and protocols used for the networks associated with business continuity, disaster recovery and continuous availability (BC/DR/CA). This article focuses in-depth on one of those protocols—Fibre Channel over Internet Protocol (FCIP)—and how it's used in a mainframe environment to provide long-distance extension networks between data centers.

Because of the higher costs of long-distance dark fiber connectivity compared with other communications services, use of the more common and more affordable IP network services is an attractive option for FC extension between geographically separated data centers. FCIP is a technology for interconnecting FC-based storage networks over distance using IP (see Figure 1).

FCIP Basics

IP storage initiatives have evolved on a foundation of previously established standards for Ethernet and IP. For example, Ethernet is standardized in IEEE 802.3, Gigabit Ethernet (GbE) is standardized in IEEE 802.3z and 10 GbE is standardized in IEEE 802.3a. IP-related standards are established by the Internet Engineering Task Force (IETF) through a diverse set of request for comments (RFCs) that cover a wide range of protocol management issues. Transmission Control Protocol (TCP) was standardized in 1981 as RFC 793, User Datagram Protocol (UDP) was standardized in 1980 as RFC 768 and FCIP was standardized in 2004 as RFC 3821.

Ethernet and IP are only half the equation for IP storage networks. IP storage technology must also accommodate previously established standards for Small Computer System Interface (SCSI), which is the purview of the InterNational Committee for Information Technology Standards (INCITS) T10. FCIP storage solutions must also follow the previously established standards for Fibre Channel Protocol (FCP) (INCITS T10) and Fibre Channel Transport (INCITS T11).

These standards all provide guideposts for technology development with the ideal goal of product interoperability. Guideposts aren't intended to be moved; in other words, new technology developments shouldn't require changes to existing standards.

RFC 3821

The RFC 3821 specification is a 75-page document that describes mechanisms that allow the interconnection of islands of FC storage area networks (SANs) to form a unified SAN in a single fabric via connectivity between islands over IP. The chief motivation behind defining these interconnection mechanisms is a desire to connect physically remote FC sites, allowing remote disk/DASD access, tape backup and live/real-time mirroring of storage between remote sites. The FC standards have chosen nominal distances between switch elements that are less than the distances available in an IP network. Since FC and IP networking technologies are compatible, it's logical to turn to IP networking for extending the allowable distances between FC switch elements.

The fundamental assumption made in RFC 3821 that you should remember is that FC traffic is carried over the IP network in such a manner that the FC fabric and all the FC devices in the fabric are unaware of the presence of the IP network. This means the FC datagrams must be delivered in such time as to comply with the existing FC specifications. The FC traffic may span LANs, metropolitan area networks (MANs) and wide area networks (WANs) as long as this fundamental assumption is adhered to.

FCIP Fundamental Concepts

FCIP is a technology for interconnecting FC-based storage networks over extended distances via IP networks. FCIP enables an end user to use his existing IP WAN infrastructure to connect FC SANs. FCIP is a means of encapsulating FC frames within TCP/IP and sending these IP packets over an IP-based network specifically for linking FC SANs over these WANs. FCIP implements tunneling techniques to carry the FC traffic over the IP network. The tunneling is Upper Layer Protocol (ULP) transparent; i.e., both FICON and FCP traffic can be sent via these FCIP tunnels over an IP-based network.

FCIP supports applications such as RDR, centralized SAN backup and data migration over very long distances that are impractical or costly using native FC connections. FCIP tunnels, built on a physical connection between two extension switches or blades, allow FC I/O to pass through the IP WAN.

The TCP connections ensure in-order delivery of FC frames and lossless transmission. The FC and all FC targets and initiators are unaware of the presence of the IP WAN. Figure 2 shows the relationship of the FC and TCP/IP layers and the general concept of FCIP tunneling.

IP addresses and TCP connections are used only at the FCIP tunneling devices at each endpoint of the IP WAN "cloud." Each IP network connection on either side of the WAN cloud is identified by an IP address and a TCP/IP connection between the two FCIP devices. An FCIP data engine encapsulates the entire FC frame in TCP/IP and sends it across the IP network (WAN). At the receiving end, the IP and TCP headers are removed and a native FC frame is delivered to the destination FC node. The existence of the FCIP devices and IP WAN cloud is transparent to the FC switches, and the FC content buried in the IP datagram is transparent to the IP network. TCP is required for transit across the IP tunnel to enforce in-order delivery of data and congestion control. The two FCIP devices in Figure 1 use the TCP connection to form a virtual interswitch link (ISL) between them known as a VE_Port. They can pass FC Class F traffic as well as data over this virtual ISL.

Let's look at a simple real-world analogy. This FCIP process is similar to writing a letter in one language and putting it into an envelope to be sent through the postal system to some other destination in the world. At the receiving end, the envelope is opened and the contents of the letter are read. Along the route, those handling the

letter don't have to understand the contents of the envelope, only where its intended destination is and where it came from. FCIP takes FC frames regardless of what the frame is for (FICON, FCP, etc.) and places these into IP frames (envelopes) for transmission to the receiving destination. At the receiving destination, the envelopes are opened and the contents are placed back on the FC network to continue their trip. In Figure 2, FC frames carrying FICON, FCP and other ULP traffic are simply sent from the SAN on the left to the SAN on the right. The frames are placed into IP wrapper packets and transmitted over the IP WAN that could be using 10/100 Gb, 10 GbE, Synchronous Optical Networking/Synchronous Digital Hierarchy (SONET/SDH) or even Asynchronous Transfer Mode (ATM) as underlying interfaces/protocols.

FCIP Terminology

Tunnels. An FCIP tunnel carries FC traffic (frames) over IP networks in such a way that the FC fabric and all FC devices in the fabric are unaware of the IP network's presence. FC frames "tunnel" through IP networks by dividing frames, encapsulating the result in IP packets upon entering the tunnel and then reconstructing them as they leave the tunnel. FCIP tunnels are used to pass FC I/O through an IP network. FCIP tunnels are built on a physical connection between two peer switches or blades. An FCIP tunnel forms a single, logical tunnel from the circuits. A tunnel scales bandwidth with each added circuit, providing lossless recovery during path failures and ensuring in-order frame delivery. You can configure an FCIP tunnel by specifying a VE_Port for a source and destination interface. When you configure the tunnel, you will provide the IP address for the source destination IP interface.

Once a TCP connection is established between FCIP entities, a tunnel is established. The two FCIP platforms at the FCIP tunnel endpoints establish a standard FC ISL through this tunnel. Each end of the FCIP tunnel appears to the IP network as a server, not as a switch. The FCIP tunnel itself appears to the switches to be just a cable. 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. Figure 3 shows an example of FCIP tunnels.

FCIP interfaces. You must configure unique IP interfaces (IPIFs) on each switch or blade GbE port used for FCIP traffic. An IPIF consists of an IP address, netmask and a maximum transmission unit (MTU) size. If the destination FCIP interface isn't on the same subnet as the GbE port IP address, you must configure an IP route to that destination. A port can contain multiple IP interfaces.

Circuits and metrics. FCIP circuits are the building blocks for FCIP tunnels and FCIP trunking. Circuits provide the links for traffic flow between source and destination FCIP interfaces that are located on either end of the tunnel. For each tunnel, you can configure a single circuit or a trunk consisting of multiple circuits. Each circuit is a connection between a pair of IP addresses associated with source and destination endpoints of an FCIP tunnel. In other words, the circuit is an FCIP connection between two unique IP

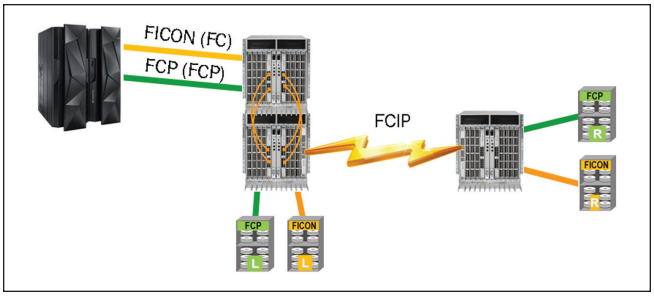

Figure 1: FCIP Passes FCP and FICON I/O Traffic Across Long-Distance IP Links

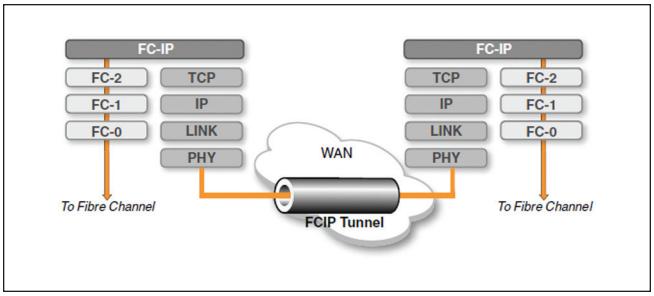

Figure 2: FCIP Tunnel Concept and TCP/IP Layers

addresses. An Ethernet interface can have one or more FCIP circuits, each requiring a unique IP address. Circuits in a tunnel can use the same or different Ethernet interfaces. Each circuit automatically creates multiple TCP connections that can be used with quality of service (QoS) prioritization. Figure 4 illustrates FCIP circuits.

A circuit has a "cost metric." Lower metric circuits are preferred over higher metric circuits. When there are circuits with different metrics, all traffic goes through the circuits with the lowest metric and no traffic goes through circuits with a higher metric. If all circuits with the lowest metric fail, circuits with higher metric are used. If all circuits have the same metric, traffic flows on all circuits. The remote end of a tunnel reorders frames to maintain in-order delivery. Load-leveling is automatically done across circuits with the lowest metric. Multiple circuits can be configured per GbE port by assigning them unique IPIFs. When you configure a circuit, you will provide the IP addresses for its source and destination interfaces.

VE_Ports. Special types of ports, known as virtual expansion ports (VE_Ports), function somewhat like an E_Port. The link between VE_Ports is called an ISL. FCIP tunnels emulate FC ports on the extension switch or blade at each end of the tunnel. Once the FCIP tunnels are configured and the TCP connections are established for a complete FCIP circuit, an ISL is activated between the switches. Once the tunnel and ISL connection are established between the switches, these logical FC ports appear as VE_Ports. VE_Ports operate like FC E_Ports for all fabric services and fabric operating system operations. Rather than using FC as the underlying transport, however, VE_Ports use TCP/IP over GbE or 10 GbE. An FCIP tunnel is assigned to a VE_Port on the switch or blade at each end of the tunnel. Since multiple VE_Ports can exist on an extension switch or blade, you can create multiple tunnels

through the IP network. FC frames enter FCIP through VE_Ports and are encapsulated and passed to TCP layer connections. An FCIP complex (FCIP Data Engine) on the switch or blade handles the FC frame encapsulation, de-encapsulation and transmission to the TCP link.

Internal to the FCIP extension device are application-specific integrated circuits (ASICs), which understand only the FC protocol. The VE_Ports are logical representations of actual FC ports on those ASICs. Consider the VE_Ports as the transition point from the FC world to the TCP/IP world inside these devices. In actuality, multiple FC ports "feed" a VE_Port (see Figure 5).

FCIP trunking. An FCIP trunk is a tunnel consisting of multiple FCIP circuits. FCIP trunking provides multiple source and destination addresses for routing traffic over a WAN, allowing load leveling, failover, failback, in-order delivery and bandwidth aggregation capabilities. FCIP trunking allows you to manage WAN bandwidth and provides redundant paths over the WAN that can protect against transmission loss due to WAN failure. FCIP trunking also provides granular load balancing on a weighted round-robin basis per batch. Trunking is enabled by creating logical circuits within an FCIP tunnel so the tunnel utilizes multiple circuits to carry traffic between multiple source and destination addresses.

FCIP trunking provides these benefits:

- Single logical tunnel comprised of one or more individual circuits
- Efficient use of VE_Ports
- Aggregation of circuit bandwidth
- Failover
- Failover metrics
- Use of disparate characteristic WAN paths
- Lossless link loss (LLL)
- In-order delivery (IOD)
- Non-disruptive link loss
- Single termination point for protocol acceleration.

FCIP trunking in essence provides a single logical tunnel comprised of multiple circuits. A single-circuit FCIP tunnel is referred to as an FCIP tunnel. An FCIP tunnel with multiple circuits is referred to as an FCIP trunk, simply because multiple circuits are being trunked together. An FCIP tunnel or FCIP trunk is a single ISL and should be treated as such in SAN fabric designs. Circuits are individual FCIP connections within the trunk, each with its own unique source and destination IP address. Because an FCIP tunnel is an ISL, each end requires its own VE_Port. If each circuit is its own FCIP tunnel, a VE_Port is required for each circuit, but with FCIP trunking each circuit isn't its own FCIP tunnel, hence the term "circuit." Since an FCIP trunk is logically a single tunnel, only a single VE or VEX_Port is used, regardless of the fact that more than one circuit may be contained within the tunnel.

Figure 6 shows an example of two FCIP tunnels that are trunking four circuits in tunnel 1 and two circuits in tunnel 2. Each circuit has been assigned a unique IP address by way of virtual IP interfaces within the FCIP

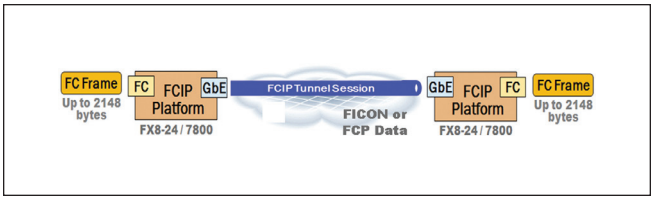

Figure 3: An FCIP Tunnel Example

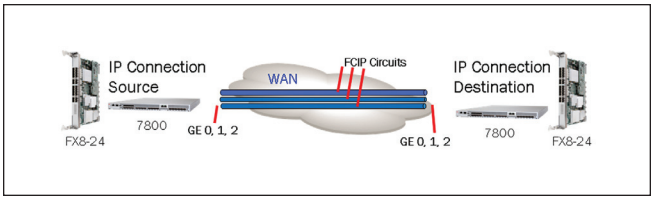

Figure 4: FCIP Circuits

device. Those IP interfaces are, in turn, assigned to Ethernet interfaces. In this case, each IP interface has been assigned to a different Ethernet interface. This isn't required, however. Ethernet interface assignment is flexible, depending on the environment's needs, and assignments can be made as desired. For instance, multiple IP interfaces can be assigned to a single Ethernet interface. The circuit flows from IP interface to IP interface through the assigned Ethernet interfaces.

TCP's Role in FCIP Extension Networks

TCP is a standard protocol described by IETF RFC 793. Unlike UDP, which is connectionless, TCP is a connection-oriented transport protocol that guarantees reliable in-order delivery of a stream of bytes between the endpoints of a connection. TCP connections ensure in-order delivery of FC frames, error recovery and lossless transmission in our FCIP extension networks.

TCP achieves this by assigning each byte of data a unique sequence number, maintaining timers, acknowledging received data through the use of acknowledgements (ACKs) and retransmission of data, if necessary. Once a connection is established between the endpoints, data can be transferred. The data stream that passes across the connection is considered a single sequence of 8-bit bytes, each of which is given a sequence number. TCP doesn't assume reliability from the lower level protocols (such as IP), so TCP must guarantee this itself.

TCP can be characterized by the following facilities it provides for applications using it:

- **Stream data transfer.** From an application's viewpoint, TCP transfers a contiguous stream of bytes through the network. The application doesn't have to bother with chopping the data into basic blocks or datagrams. TCP does this by grouping the bytes in TCP segments, which are passed to IP for transmission to the destination. TCP itself decides how to segment the data and can forward the data at its own convenience.
- **Flow control.** The receiving TCP, when sending an ACK back to the sender, also indicates to the sender the number

FCIP is a technology for interconnecting FC-based storage networks over extended distances via IP networks.

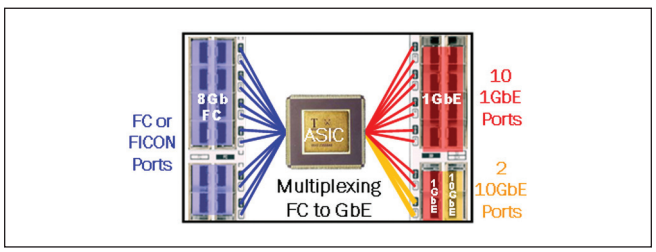

Figure 5: ASIC and VE_Port Concept

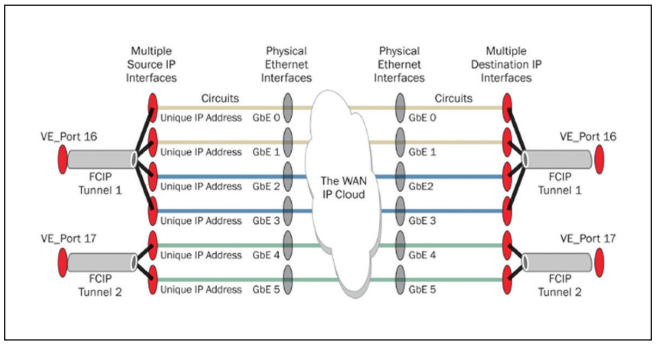

Figure 6: FCIP Tunnels and Their IP/Ethernet Interfaces and Circuits

of bytes it can receive beyond the last received TCP segment without causing overrun and overflow in its internal buffers. This is sent in the ACK in the form of the highest sequence number it can receive without problems. This is also known as the TCP window mechanism.

- **Reliability.** TCP assigns a sequence number to each byte transmitted and expects a positive ACK from the receiving TCP. If the ACK isn't received within a timeout interval, the data is retransmitted. Since the data is transmitted in blocks (TCP segments), only the sequence number of the first data byte in the segment is sent to the destination host. The receiving TCP uses the sequence numbers to rearrange the segments when they arrive out of order and eliminate duplicate segments.
- **Logical connections.** The reliability and flow control mechanisms described previously require that TCP initializes and maintains certain status information for each datastream. The combination of this status, including sockets, sequence numbers and window sizes, is called a logical connection in TCP. Each such connection is uniquely identified by the pair of sockets used by the sending and receiving processes.
- **Full duplex.** TCP provides for concurrent datastreams in both directions.
- **Multiplexing.** This is achieved through the use of ports, just as with UDP.

TCP WAN Considerations and Ports

Because FCIP uses TCP connections over an existing WAN, consult with your WAN carrier and IP network administrator to ensure the network hardware and software equipment operating in the data path can properly support the TCP connections. Keep these considerations in mind:

- Routers and firewalls in the data path must be configured to pass FCIP traffic (TCP port 3225) and IPsec traffic, if IPsec is used (UDP port 500).
- To enable recovery from a WAN failure or outage, be sure that diverse, redundant network paths are available across the WAN.
- Be sure the underlying WAN infrastructure is capable of supporting the redundancy and performance expected in your implementation.

Conclusion

This follow-up article provided an overview of the various distance extension technology options and focused in-depth on one of those options: FCIP. We covered the basics of FCIP, including a look at the relevant standards, the building blocks of FCIP in our discussion of tunnels, circuits, trunking and metrics, and finally, TCP's role in an FCIP extension network. **ETJ**

Dr. Steve Guendert is a principal engineer and global solutions architect for Brocade Communications, where he leads the mainframe-related business efforts. A senior member of both the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM), he serves on the board of directors for the Computer Measurement Group (CMG). He is also a former member of the SHARE board of directors.
Email: stephen.guendert@brocade.com
Twitter: @BRCD_DrSteve