IPS Working Group
INTERNET-DRAFT
<draft-ietf-ips-fcovertcpip-06.txt>
(Expires March, 2002)
Category: standards-track

M. Rajagopal, R. Bhagwat, R. A. Helland,
LightSand Comm.
E. Rodriguez, Lucent Tech.
C. Carlson, QLogic
D. Fraser, Compaq
D. Peterson, Cisco
L. Lamers, SAN Valley
V. Chau, G. Hecht, Gadzoox Networks
S. Wilson, B. Snively, R. Weber, Brocade Comm.
M. O'Donnell, A. Rijhsinghani, McDATA
S. Rupanagunta, Aarohi Comm.
V. Rangan, Rhapsody Networks
J. Nelson, K. Hirata, Vixel
M. Merhar, Pirus Networks
N. Wanamaker, Akara

# Fibre Channel Over TCP/IP (FCIP)

## Status of this Memo



## Abstract

Fibre Channel Over TCP/IP (FCIP) describes mechanisms that allow the interconnection of islands of Fibre Channel storage area networks over IP-based networks to form a unified storage area network in a single Fibre Channel fabric. FCIP relies on IP-based network services to provide the connectivity between the storage area network islands over local area networks, metropolitan area networks, or wide area networks.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [2].

Table Of Contents

## 1. Purpose, Motivation and Objectives

Fibre Channel (FC) is a gigabit speed networking technology primarily used to implement Storage Area Networks (SANs). See section 2 for information about how Fibre Channel is standardized and the relationship of this specification to Fibre Channel standards.

This specification describes mechanisms that allow the interconnection of islands of Fibre Channel SANs over IP Networks to form a unified SAN in a single Fibre Channel fabric. The motivation behind defining these interconnection mechanisms is a desire to connect physically remote FC sites allowing remote disk access, tape backup, and live mirroring.

Fibre Channel standards have chosen nominal distances between switch

elements that are less than the distances available in an IP Network. Since Fibre Channel and IP Networking technologies are compatible, it is logical to turn to IP Networking for extending the allowable distances between Fibre Channel switch elements.

The fundamental assumption made in this specification is that the Fibre Channel traffic is carried over the IP Network in such a manner that the Fibre Channel Fabric and all Fibre Channel devices on the Fabric are unaware of the presence of the IP Network. This means that the FC datagrams MUST be delivered in such time as to comply with existing Fibre Channel specifications. The FC traffic





MAY span LANs, MANs and WANs, so long as this fundamental assumption is adhered to.

The objectives of this document are to:

1) specify the encapsulation and mapping of Fibre Channel (FC) frames employing FC Frame Encapsulation [25].

2) apply the mechanism described in 1) to an FC Fabric using an IP network as an interconnect for two or more islands in an FC Fabric.

3) address any FC concerns arising from tunneling FC traffic over an IP-based network, including security, data integrity (loss), congestion, and performance. This will be accomplished by utilizing the existing IETF-specified suite of protocols.

4) be compatible with the referenced FC standards. While new work may be undertaken in T11 [7] to optimize and enhance FC Fabrics, this specification requires conformance only to the referenced FC standards.

5) be compatible with all applicable IETF standards so that the IP Network used to extend an FC Fabric can be used concurrently for other reasonable purposes.

## 2. Relationship to Fibre Channel Standards

### 2.1 Relevant Fibre Channel Standards

FC is standardized under American National Standard for Information

Systems of the National Committee for Information Technology Standards (ANSI-NCITS) in its T11 technical committee. T11 has specified a number of documents describing FC protocols, operations, and services. T11 documents of interest to readers of this specification include (but are not limited to):

- FC-BB - Fibre Channel Backbone [3]
- FC-BB-2 - Fibre Channel Backbone -2 [4]
- FC-SW-2 - Fibre Channel Switch Fabric -2 [5]
- FC-FS - Fibre Channel Framing and Signaling [6]

FC-BB and FC-BB-2 describe the relationship between an FC Fabric and interconnect technologies not defined in by Fibre Channel standards (e.g., ATM and SONET). FC-BB-2 is the natural Fibre Channel home for describing relationships to TCP/IP and FCIP.





FC-SW-2 describes the switch components of an FC Fabric and FC-FS describes the FC Frame format and basic control features of Fibre Channel.

Additional information regarding T11 activities is available on the committee's web site [7].

## 2.2 This Specification and Fibre Channel Standards

When considering the challenge of transporting FC Frames over an IP Network, it is logical to divide the standardization effort between TCP/IP requirements and Fibre Channel requirements. This specification covers the TCP/IP requirements for transporting FC Frames and the Fibre Channel documents described in section 2.1 cover the Fibre Channel requirements.

This specification addresses only the requirements necessary to properly utilize an IP Network as a conduit for FC Frames. The result is a specification for an FCIP Entity (see section 5.4).

A product that tunnels an FC Fabric through an IP Network must combine the FCIP Entity with an FC Entity (see section 5.3) using an implementation specific interface. The requirements placed on an FC Entity by this specification to achieve proper delivery of FC Frames are summarized in annex E. More information about FC Entities can be

found in the Fibre Channel standards and an example of an FC Entity can be found in FC-BB-2 [4].

No attempt is being made to define a specific API between an FCIP Entity and an FC Entity at this time because doing so risks compromising the performance and efficacy of the resulting products. Current experience in this area is simply insufficient to guide definition of the interface appropriately.

The objectives and motivations of this specification are not impacted by the decision not to standardize a specific API between FCIP Entities and FC Entities because fully functional and compliant products can be built provided they contain both an FCIP Entity and an FC Entity. The only products that cannot be built are those that contain only one or the other.

## 3. Terminology

Terms needed to clarify the concepts presented in FCIP are defined here.

FC End Node - A FC device that uses the connection services provided by the FC Fabric.





FC Entity - The Fibre Channel specific element that combines with an FCIP Entity to form an interface between an FC Fabric and an IP Network (see section 5.3).

FC Fabric - An entity that interconnects various Nx_Ports (see [6]) attached to it, and is capable of routing FC Frames using only the destination ID information in a FC Frame header (see annex C).

FC Frame - The basic unit of Fibre Channel data transfer (see annex C).

FC Receiver Portal - The access point through which an FC Frame and time stamp enters an FCIP Data Engine from the FC Entity.

FC Transmitter Portal - The access point through which a reconstituted FC Frame and time stamp leaves an FCIP Data Engine to the FC Entity.

FCIP Data Engine (FCIP_DE) - The component of an FCIP Entity that handles FC Frame encapsulation, de-encapsulation, and transmission FCIP Frames through a single TCP connection (see section 5.6).

FCIP Entity - The principal FCIP interface point to the IP Network (see section 5.4).

FCIP Frame - An FC Frame plus the FC Frame Encapsulation [25] header and encoded EOF that contains the FC Frame (see section 5.6.1).

FCIP Link - One or more TCP connections that connect one FCIP_LEP to another (see section 5.2).

FCIP Link Endpoint (FCIP_LEP) - The component of an FCIP Entity that handles FC Frame encapsulation, de-encapsulation, and transmission through a single FCIP Link (see section 5.5).

Encapsulated Frame Receiver Portal - The TCP access point through which an FCIP Frame is received from the IP Network by an FCIP Data Engine.

Encapsulated Frame Transmitter Portal - The TCP access point through which an FCIP Frame is transmitted to the IP Network by an FCIP Data Engine.





## 4. Protocol Summary

The FCIP protocol is summarized as follows:

1) The primary function of an FCIP Entity is forwarding FC Frames, employing FC Frame Encapsulation described in [25].

2) Viewed from the IP Network perspective, FCIP Entities are peers and communicate using TCP/IP. Each FCIP Entity is a TCP endpoint in the IP-based network.

3) Viewed from the FC Fabric perspective, pairs of FCIP Entities,

in combination with their associated FC Entities, serve as an FC Frame transmission component of the FC Fabric. The FC End Nodes are unaware of the existence of the FCIP Link.

4) FC Primitive Signals, Primitive Sequences, and Class 1 FC Frames are not transmitted across an FCIP Link because they cannot be encoded using FC Frame Encapsulation [25].

5) The path (route) taken by an encapsulated FC Frame follows the normal routing procedures of the IP Network.

6) At any instant in time, an FCIP Entity SHALL NOT have more than one IP Address.

7) An FCIP Entity may contain multiple FCIP Link Endpoints, but each FCIP Link Endpoint (FCIP_LEP) communicates with exactly one other FCIP_LEP.

8) When multiple FCIP_LEPs with multiple FCIP_DEs are in use, selection of which FCIP_DE to use for encapsulating and transmitting a given FC Frame is outside the scope of this document. FCIP Entities do not actively participate in FC Frame routing.

9) The FCIP Control & Services function MAY use TCP/IP quality of service features (see section 9.2) to support Fibre Channel capabilities.

10) Each FCIP Entity is statically or dynamically configured with a list of IP addresses and port numbers corresponding to participating FCIP Entities. If dynamic discovery of participating FCIP Entities is supported, the function SHALL be performed using the Service Location Protocol (SLPv2) [23]. It is outside the scope of this specification to describe any static configuration method for participating FCIP Entity





discovery. Refer to section 7.1.4 for a detailed description of dynamic discovery of participating FCIP Entities using SLPv2.

11) FCIP Entities do not actively participate in the discovery of FC source and destination identifiers. Discovery of FC addresses (accessible via the FCIP Entity) is provided by techniques and protocols within the FC architecture as described in FC-FS [6]

and FC-SW-2 [5].

12) To support IP Network security (see section 8), FCIP Entities MUST:
    1) implement cryptographically protected authentication and cryptographic data integrity keyed to the authentication process, and
    2) implement data confidentiality security features.

13) On a given TCP connection, this specification relies on TCP/IP to deliver a byte stream in the same order that it was sent.

14) This specification relies on both TCP and FC error recovery mechanisms to detect and recover from data loss and corruption within the IP Network.





## 5. The FCIP Model

## 5.1 FCIP Protocol Model

The relationship between FCIP and other protocols is illustrated in figure 1.

```
+------------------------+ FCIP Link +------------------------+
|          FCIP          |===========|          FCIP          |
+---------+------+---------+         +---------+------+---------+
|  FC-2   |      |   TCP   |         |   TCP   |      |  FC-2   |
+---------+      +---------+         +---------+      +---------+
|  FC-1   |      |   IP    |         |   IP    |      |  FC-1   |
+---------+      +---------+         +---------+      +---------+
|  FC-0   |      |  LINK   |         |  LINK   |      |  FC-0   |
+---------+      +---------+         +---------+      +---------+
     |           |   PHY   |         |   PHY   |           |
     |           +---------+         +---------+           |
     |                |                   |                |
     |                |    IP Network     |                |
     V                +-------------------+                V
  to Fibre                                              to Fibre
  Channel                                               Channel
Environment                                           Environment

Fig. 1  FCIP Protocol Stack Model
```

Note that the objective of the FCIP Protocol is creation and maintenance of one or more FCIP Links.

## 5.2 FCIP Link

The FCIP Link is the basic unit of service provided by the FCIP Protocol to an FC Fabric. As shown in figure 2, an FCIP Link connects two portions of an FC Fabric using an IP Network as a transport to form a single FC Fabric.

```
/\/\/\/\/\/\         /\/\/\/\/\/\         /\/\/\/\/\/\
\     FC     /   FCIP  \    IP     /   Link  \     FC     /
/   Fabric   \=========/  Network \=========/   Fabric   \
\/\/\/\/\/\/         \/\/\/\/\/\/         \/\/\/\/\/\/

Fig. 2  FCIP Link Model
```

At the points where the ends of the FCIP Link meet portions of the FC Fabric, an FCIP Entity (see section 5.4) combines with an FC Entity as described in section 5.3 to serve as the interface between FC and IP.

An FCIP Link SHALL contain at least one TCP connection and MAY contain more than one TCP connection. The endpoints of a single TCP connection are FCIP Data Engines (see section 5.6). The endpoints of a single FCIP Link are FCIP Link Endpoints (see section 5.5).

## 5.3 FC Entity

A product that tunnels an FC Fabric through an IP Network must combine an FC Entity with an FCIP Entity (see section 5.4) to form a complete interface between the FC Fabric and IP Network as shown in figure 3.

```
+-----------+         /\/\/\/\/\/\         +-----------+
|   FCIP    |   FCIP  \     IP     /   Link  |   FCIP    |
|  Entity   |=========/  Network \=========|  Entity   |
+-----------+         \/\/\/\/\/\/         +-----------+
|    FC     |                              |    FC     |
|  Entity   |                              |  Entity   |
+-----------+                              +-----------+
     |                                          |
/\/\/\/\/\/\                               /\/\/\/\/\/\
\    FC    /                               \    FC    /
/  Fabric  \                               /  Fabric  \
\/\/\/\/\/\/                               \/\/\/\/\/\/
```

Fig. 3  FC Entity and FCIP Entity Model

In general, the combination of an FCIP Link and FC and FCIP Entities is intended to replace a Fibre Channel defined connection between Fibre Channel components. For example, this combination can be used to replace a hard-wire connection between two Fibre Channel switches. There are limitations on the generally intended usage of the combination shown in figure 3. As another example, the combination cannot be used to replace cable connections in a Fibre Channel Arbitrated Loop because loop primitive signals cannot be encapsulated for transmission over TCP.

The interface between the FC and FCIP Entities is implementation specific. The minimum requirements placed on an FC Entity by this specification are listed in annex E. More information about FC Entities can be found in the Fibre Channel standards and an example of an FC Entity can be found in FC-BB-2 [4].

## 5.4 FCIP Entity

The model for an FCIP Entity is shown in figure 4.

```
       .......................................................
       : FCIP Entity                                         :
       :                                                     :
       :  +-----------+                                      :
       :  | FCIP      |                                      :
       :  | Control & |------------------------------------+ :
       :  | Services  |                                    | :
       :  | Module    |                                    | :
       :  +-----------+                                    | :
       :        |            +--------------------+        | :
       :        |   +-------+--------------------+|----+   | :
       :        |   |+-----+--------------------+|----+|   | :
       :        |   ||+----| FCIP Link Endpoint |----+||   | :
       :        |   |||    +--------------------+    |||   | :
       :.............................................|||.....:
                |   |||                              |||   |
                |   |||                              |||   o<--+
                |   |||                unique TCP    |||   |   |
                |   |||                connections-->|||   |   |
                |   |||                              |||   |   |
             +----------+                         /\/\/\/\/\/\ |
             |    FC    |                         \    IP    / |
             |  Entity  |                         /  Network \ |
             +----------+                         \/\/\/\/\/\/ |
                  |                                            |
             /\/\/\/\/\/\                   +------------------+
             \    FC    /                   +->IP Address &
             /  Fabric  \                   +->Well Known Port
             \/\/\/\/\/\/
```

Fig. 4  FCIP Entity Model

The FCIP Entity is the connection interface point for the IP Network and is the owner of the IP Address and Well Known Port used to form TCP connections. An FC Fabric to IP Network interface product SHALL contain one FCIP Entity for each IP Address assigned to the product.

An FCIP Entity contains an FCIP Control & Services Module to provide the FC Entity with an interface to key IP Network features. The interfaces to the IP Network features is implementation specific, however, to maintain interoperability, the TCP/IP mechanisms used are specified in this document as follows:

- TCP Connections - see section 7





- Security - see section 8
- Performance - see section 9
- Dynamic Discovery - see section 7.1.4

The FCIP Link Endpoints in an FCIP Entity provide the FC Frame encapsulation and transmission features of FCIP.

5.5 FCIP Link Endpoint (FCIP_LEP)

Each time a TCP connection is formed to an IP Address for which no TCP connection already exists, the FCIP Entity SHALL create a new FCIP Link Endpoint containing one FCIP Data Engine.

An FCIP_LEP is a transparent data translation point between an FC Entity and an IP Network. A pair of FCIP_LEPs communicating over one or more TCP connections create an FCIP Link to join two islands of a FC Fabric, producing a single FC Fabric.

The IP Network over which the two FCIP_LEPs communicate is not aware of the FC payloads that it is carrying. Likewise, the FC End Nodes connected to the FC Fabric are unaware of the TCP/IP based transport employed in the structure of the FC Fabric.

As shown in figure 5, the FCIP Link Endpoint contains one FCIP Data Engine for each TCP connection in the FCIP Link.

```
...............................................
: FCIP Link Endpoint                          :
:                  +-------------------+      :
:         +-------+-------------------+|----+ :
:         |+-----+-------------------+|----+| :
:         ||+----| FCIP Data Engine |----+|| :
:         |||    +-------------------+    ||| :
:.............................................:
          |||                             |||
```

```
          +----------+                     /\/\/\/\/\/\
          |    FC    |                     \    IP    /
          |  Entity  |                     /  Network \
          +----------+                     \/\/\/\/\/\/
                |
          /\/\/\/\/\/\
          \    FC    /
          /  Fabric  \
          \/\/\/\/\/\/
```

Fig. 5  FCIP Link Endpoint Model





An FCIP_LEP uses normal TCP based flow control mechanisms for managing its internal resources and matching them with the advertised TCP Receiver Window Size (see section 7.5). An FCIP_LEP MAY communicate with its FC Entity counterpart to coordinate flow control.

## 5.6 FCIP Data Engine (FCIP_DE)

The model for one of the multiple FCIP_DEs that may be present in an FCIP_LEP is shown in figure 6.

```
          +--------------------------------+
          |                                |
          |-+    +------------------+    +-|
     C    |p|    |  Encapsulation   |    |p|    N
   F h -->|1|--->|      Engine      |--->|2|--> e
   i a    |-+    +------------------+    +-|    t
   b n    |                                |  I w
   r n    |-+    +------------------+    +-|  P o
   e e    |p|    | De-Encapsulation |    |p|    r
     l <--|4|<---|      Engine      |<---|3|<-- k
          |-+    +------------------+    +-|
          |                                |
          +--------------------------------+
```

Fig. 6  FCIP Data Engine Model

Data enters and leaves the FCIP_DE through four portals (p1 - p4). The portals do not process or examine the data that passes through

them. They are only the named access points where the FCIP_DE interfaces with external world. The names of the portals are as follows:

p1) FC Receiver Portal - The interface through which an FC Frame and time stamp enters an FCIP_DE from the FC Entity.

p2) Encapsulated Frame Transmitter Portal - The TCP interface through which an FCIP Frame is transmitted to the IP Network by an FCIP_DE.

p3) Encapsulated Frame Receiver Portal - The TCP interface through which an FCIP Frame is received from the IP Network by an FCIP_DE.

p4) FC Transmitter Portal - The interface through which a reconstituted FC Frame and time stamp exits an FCIP_DE to the FC Entity.





The work of the FCIP_DE is done by the Encapsulation and De-Encapsulation Engines. The Engines have two functions:

1) Encapsulating and de-encapsulating FC Frames using the encapsulation format described in FC Frame Encapsulation [25] and in section 5.6.1 of this document, and

2) Detecting some data transmission errors and performing minimal error recovery as described in section 5.6.2.

Data flows through the FCIP_DE in the following seven steps:

1) An FC Frame and time stamp arrives at the FC Receiver Portal and is passed to the Encapsulation Engine. The FC Frame is assumed to have been processed by the FC Entity according to the applicable FC rules and is not validated by the FCIP_DE. If the FC Entity is in the Unsynchronized state with respect to a time base as described in the FC Frame Encapsulation [25] specification, the time stamp delivered with the FC Frame SHALL be zero.

2) In the Encapsulation Engine, the encapsulation format described in FC Frame Encapsulation [25] and in section 5.6.1 of this

document SHALL be applied to prepare the FC Frame and associated time stamp for transmission over the IP Network.

3) The entire encapsulated FC Frame (a.k.a. the FCIP Frame) SHALL be passed to the Encapsulated Frame Transmitter Portal where it SHALL be inserted in the TCP byte stream.

4) Transmission of the FCIP Frame over the IP Network follows all the TCP rules of operation. This includes but is not limited to the in-order delivery of bytes in the stream, as specified by TCP [8].

5) The FCIP Frame arrives at the partner FCIP Entity where it enters the FCIP_DE through the Encapsulated Frame Receiver Portal and is passed to the De-Encapsulation Engine for processing.

6) The De-Encapsulation Engine SHALL validate the incoming TCP byte stream as described in section 5.6.2 and SHALL de-encapsulate the FC Frame and associated time stamp according to the encapsulation format described in FC Frame Encapsulation [25] and in section 5.6.1 of this document.





7) In the absence of errors, the de-encapsulated FC Frame and time stamp SHALL be passed to the FC Transmitter Portal for delivery to the FC Entity.

Every FC Frame that arrives at the FC Receiver Portal SHALL be transmitted on the IP Network as described in steps 1 through 4 above. In the absence of errors, data bytes arriving at the Encapsulated Frame Receiver Portal SHALL be de-encapsulated and forwarded to the FC Transmitter Portal as described in steps 5 through 7.

### 5.6.1 FCIP Encapsulation of FC Frames

The FCIP encapsulation of FC Frames employs FC Frame Encapsulation [25].

The features from FC Frame Encapsulation that are unique to

individual protocols SHALL be applied as follows for the FCIP encapsulation of FC Frames.

The Protocol# field SHALL contain 1 in accordance with the IANA Considerations annex of FC Frame Encapsulation [25].

The Protocol Specific field SHALL have the format shown in figure 7. Note: the word numbers in figure 7 are relative to the complete FC Frame Encapsulation header, not to the Protocol Specific field.

```
 W|------------------------------Bit------------------------------|
 o|                                                               |
 r|3 3 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1                    |
 d|1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0|
  +---------------------------------------------------------------+
 1|               replication of encapsulation word 0             |
  +-------------------------------+-------------------------------+
 2|            reserved           |           -reserved           |
  +-------------------------------+-------------------------------+
```

Fig. 7  FCIP Usage of FC Frame Encapsulation Protocol Specific field

Word 1 of the Protocol Specific field SHALL contain an exact copy of word 0 in FC Frame Encapsulation [25].

Word 2 of the Protocol Specific field is reserved for future enhancements to the FCIP protocol.

The reserved field (bits 31-16 in word 2): SHALL contain 0.





The -reserved field (bits 15-0 in word 2): SHALL contain 65535 (or 0xFFFF).

The CRCV (CRC Valid) Flag SHALL be set to 0.

The CRC field SHALL be set to 0.

### 5.6.2 FCIP Data Engine Error Detection and Recover

#### 5.6.2.1 TCP Assistance With Error Detection and Recovery

TCP [8] REQUIRES in order delivery, generation of TCP checksums, and checking of TCP checksums. Thus, the byte stream passed from TCP to the FCIP_LEP will be in order and free of errors detectable by the TCP checksum. If TCP did not perform these functions, the FCIP_LEP would have to.

#### 5.6.2.2 Errors in FCIP Headers and Discarding FCIP Frames

Bytes delivered through the Encapsulated Frame Receiver Portal that are not correctly delimited as defined by the FC Frame Encapsulation [25] SHALL NOT be forwarded on to the FC Entity.

Synchronization of the FCIP_DE to the FCIP Frames in the data stream entering the Encapsulated Frame Receiver Portal is maintained using the FC Frame Encapsulation header's frame length field to determine where in the data stream the next FC Encapsulation header is located. Synchronization SHALL be verified on each FCIP Frame. The validity and positioning of the following FCIP Frame information SHOULD be used to verify synchronization:

a) Protocol # field and its ones complement (2 tests);
b) Version field and its ones complement (2 tests);
c) Replication of encapsulation word 0 in word 1 (1 test);
d) Reserved field and its ones complement (2 tests);
e) Flags field and its ones complement (2 tests);
f) Length field and its ones complement (2 tests);
g) CRC field is equal to zero (1 test);
h) SOF fields and ones complement fields (4 tests);
i) Format and values of FC header (1 test);
j) CRC of FC Frame (2 tests);
k) EOF fields and ones complement fields (4 tests); and/or
l) FC Frame Encapsulation header information in the next FCIP Frame (1 test).





Verification SHALL be accomplished by performing the following tests:

a) Length field validation -- 15 < Length < 545 (f above);
b) Comparison of Length field to its ones complement (f above); and
c) At least 6 other of the 22 distinct tests listed above.

Errors in FCIP Frame headers SHOULD be considered carefully, since some may be synchronization errors. For example, any failure of the Length field tests described above SHALL be handled as a synchronization error. Errors in FCIP Frames detected by the FCIP_DE that affect synchronization with the Encapsulated Frame Receiver Portal byte stream SHALL be handled as defined by section 5.6.2.3.

Whenever an FCIP_DE discards bytes delivered through the Encapsulated Frame Receiver Portal, it SHALL cause the FCIP Entity to notify the FC Entity of the condition and provide a suitable description of the reason bytes were discarded.

The burden for recovering from discarded data falls on the FC Entity and other components of the FC Fabric and is outside the scope of this specification.

5.6.2.3 Synchronization Failures

If an FCIP_DE determines that it cannot find the next FCIP Frame header in the byte stream entering through the Encapsulated Frame Receiver Portal, the FCIP_DE SHALL either:

a) close the TCP connection [8] [9];
b) recover synchronization by searching the bytes delivered by the Encapsulated Frame Receiver Portal for a valid FCIP Frame header having the correct properties, and discarding bytes delivered by the Encapsulated Frame Receiver Portal until a valid FCIP Frame header is found; or
c) attempt to recover synchronization as described in b) and if synchronization cannot be recovered close the TCP connection as described in a).

If the FCIP_DE attempts to recover synchronization, the resynchronization algorithm used SHALL meet the following requirements:

a) discard or identify with an EOFa (see annex section C.1) those FC Frames and fragments of FC Frames identified before synchronization has again been completely verified. The number of FC Frames not forwarded may vary based on the algorithm used;

b) return to sending valid FC Frames only after synchronization has been verified; and

c) close the TCP/IP connection if the algorithm ends without verifying successful synchronization. The probability of failing to synchronize successfully and the time necessary to determine whether or not synchronization was successful may vary with the algorithm used.

An example algorithm meeting these requirements can be found in annex A.

The burden for recovering from the discarding of FCIP Frames during the optional resynchronization process described in this section falls on the FC Entity and other components of the FC Fabric and is outside the scope of this specification.

## 6. Checking FC Frame Transit Times in the IP Network

The FC Entity MUST implement setup and verification components of the frame transit time function described in the FC Frame Encapsulation [25] specification to detect FC Frames that have taken too long to transit the IP Network. The choice to place this implementation requirement in the FC Entity is based on a desire to include the transit time through the FCIP Entities when computing the IP Network transit time experienced by the FC Frames.

Each FC Frame that enters the FCIP_DE through the FC Receiver Portal SHALL be accompanied by a time stamp value that the FCIP_DE SHALL place in the Time Stamp [integer] and Time Stamp [fraction] fields of the encapsulation header of the FCIP Frame that contains the FC Frame. If no synchronized time stamp value is available to accompany the entering FC Frame a value of zero SHALL be supplied.

Each FC Frame that exits the FCIP_DE through the FC Transmitter Portal SHALL be accompanied by the time stamp value taken from the FCIP Frame that encapsulated the FC Frame.

The FC Entity SHALL use suitable internal clocks and either Fibre Channel services or an SNTP Version 4 server [12] to establish and maintain the required synchronized time value. The FC Entity SHALL verify that the FC Entity it is communicating with on an FCIP Link is using the same synchronized time source as it is, either Fibre Channel services or SNTP server.

Note that since the FC Fabric is expected to have a single synchronized time value throughout, reliance on the Fibre Channel

services means that only one synchronized time value is needed for all FCIP_DEs regardless of their connection characteristics.

# 7. TCP Connection Management

## 7.1 TCP Connection Establishment

### 7.1.1 Connection Establishment Model

The description of the connection establishment process in section 7.1 is a model for the interactions between an FC Entity and an FCIP Entity during TCP connection establishment. The model is written in terms of a "shared" database that the FCIP Entity consults to determine the properties of the TCP connections to be formed combined with routine calls to the FC Entity when connections are successfully established. Whether the FC Entity contributes information to the "shared" database is not critical to this model. What is important is the fact that the FCIP Entity may consult the database at anytime to determine its actions relative to TCP connection establishment.

It is important to remember that this description is only a model for the interactions between an FC Entity and an FCIP Entity. Any implementation that has the same effects on the FC Fabric and IP Network as those described in the model meets the requirements of this specification. For example, an implementation might replace the "shared" database with a routine interface between the FC and FCIP Entities.

### 7.1.2 FCIP Entity New TCP Connection Establishment Actions

Although the establishment of a TCP connection comes about in several different ways, the FCIP Entity SHALL take the actions described in this section following successful establishment of a new TCP connection.

If the IP Address for the other end of the TCP connection is one to which no other TCP connections exist, the FCIP Entity SHALL:

1) Instantiate the appropriate Quality of Service (see section 9) conditions on the newly created TCP connection,

2) Create a new FCIP_LEP for the new FCIP Link,

3) Create a new FCIP_DE within the newly created FCIP_LEP to service the new TCP connection, and

4) Inform the FC Entity of the new FCIP_LEP and FCIP_DE.





If the IP Address for the other end of the TCP connection is one for which a TCP connection already exists, the FCIP Entity SHALL:

1) Instantiate the appropriate Quality of Service (see section 9) conditions on the newly created TCP connection,

2) Create a new FCIP_DE within the existing FCIP_LEP to service the new TCP connection, and

3) Inform the FC Entity of the FCIP_LEP and new FCIP_DE.

### 7.1.3 Non-Dynamic Creation of New TCP Connections

When an FCIP Entity discovers that a new TCP Connection needs to be established, it SHALL establish all enabled IP security features as described in section 8 and then determine the following information about the new connection:

- IP Address
- TCP Connection Parameters (see section 7.3)
- Security Information (see section 8)
- Quality of Service Information (see section 9)

Based on this information, the FCIP Entity SHALL generate a TCP connect request [8] to the FCIP Well-Known Port at the specified IP Address. If the TCP connect request is rejected, the FCIP Entity SHALL act to limit unnecessary repetition of attempts to establish similar connections. If the TCP connect request is accepted, the FCIP Entity finalize the connection setup as described in section 7.1.2.

### 7.1.4 Dynamic Creation of New TCP Connections

If dynamic discovery of participating FCIP Entities is supported the function SHALL be performed using the Service Location Protocol (SLPv2) [23] in the manner defined for FCIP usage [26].

Upon discovering that dynamic discovery is to be used, the FCIP Entity SHALL establish all enabled IP security features as described in section 8 and then:

1) Determine the one or more FCIP Discovery Domain(s) to be used in the dynamic discovery process;

2) Establish an SLPv2 Service Agent to advertise the availability of this FCIP Entity to peer FCIP Entities in the identified FCIP Discovery Domain(s); and





3) establish an SLPv2 User Agent to locate service advertisements for peer FCIP Entities in the identified FCIP Discovery Domain(s).

For each peer FCIP Entity dynamically discovered through the SLPv2 User Agent, the FCIP Entity SHALL determine the following information about the new connection:

- TCP Connection Parameters (see section 7.3)
- Security Information (see section 8)
- Quality of Service Information (see section 9)

Based on this information, the FCIP Entity shall generate a TCP connect request [8] to the FCIP Well-Known Port at the IP Address specified by the service advertisement. If the TCP connect request is rejected, act to limit unnecessary repetition of attempts to establish similar connections. If the TCP connect request is accepted, the FCIP Entity finalize the connection setup as described in section 7.1.2.

### 7.1.5 Processing Incoming TCP Connect Requests

The FCIP Entity SHALL listen for new TCP connection requests [8] on the FCIP Well-Known Port. An FCIP Entity MAY also accept and establish TCP connections to a TCP port number other than the FCIP Well-Known Port, as configured by the network administrator.

The FCIP Entity SHALL determine the following information about the requested connection:

- Whether the requested connection is allowed

- Whether IP security setup has been performed for the IP security features enabled on the connection (see section 8)
- Quality of Service Information (see section 9)

If the requested connection is not allowed, the FCIP Entity SHALL terminate the TCP connect request [8]. If the requested connection is allowed, the FC Entity SHALL accept the TCP connect request and setup the local definition of the new TCP Connection as described in section 7.1.2.

## 7.2 Closing TCP Connections

The FCIP Entity SHALL provide a mechanism by which the FC Entity is able to cause the closing of an existing TCP Connection at anytime. This allows the FC Entity to close TCP connections that are producing too many errors, etc.





## 7.3 TCP Connection Parameters

In order to provide efficient management of FCIP_LEP resources as well as FCIP Link resources, consideration of certain TCP connection parameters is RECOMMENDED.

### 7.3.1 TCP Selective Acknowledgement Option

The Selective Acknowledgement option RFC 2883 [24] allows the receiver to acknowledge multiple lost packets in a single ACK, enabling faster recovery. An FCIP Entity MAY negotiate use of TCP SACK and use it for faster recovery from lost packets and holes in TCP sequence number space.

### 7.3.2 TCP Window Scale Option

This option allows TCP window sizes larger than 16-bit limits to be advertised by the receiver. It is necessary to allow data in long fat networks to fill the available pipe. This also implies buffering on the TCP sender that matches the (bandwidth*delay) product of the TCP connection. An FCIP_LEP uses locally available mechanisms to set a window size that matches the available local buffer resources and the desired throughput.

### 7.3.3 Protection against sequence number wrap

It is RECOMMENDED that FCIP Entities implement protection against sequence number wrap. It is quite possible that within a single connection, TCP sequence numbers wrap within a timeout window.

#### 7.3.4 TCP No Delay Option

FCIP Entities SHALL disable the Nagle TCP No Delay option. This option is designed for usage in a telnet environment.

### 7.4 TCP Connection Considerations

In idle mode, a TCP connection "keep alive" option of TCP is normally used to keep a connection alive. However, this timeout is fairly large and may prevent early detection of loss of connectivity. In order to facilitate faster detection of loss of connectivity, FC Entities SHOULD implement some form of Fibre Channel connection failure detection.

When an FCIP Entity discovers that a TCP connectivity has been lost, the FCIP Entity SHALL notify the FC Entity of the failure.





### 7.5 Flow Control Mapping between TCP and FC

The FCIP Entity and FC Entity are connected to the IP Network and FC Fabric, respectively, and they need to follow the flow control mechanisms of both TCP and FC, which work independent of each other.

This section provides guidelines as to how the FCIP Entity can map TCP flow control to status notifications to the FC Entity.

There are two scenarios when the flow control management becomes crucial:

1) When there is line speed mismatch between the FC and IP interfaces.

   Even though it is RECOMMENDED that both the FC and IP interfaces to the FC Entity and FCIP Entity, respectively, be of comparable speeds, it is possible to carry FC traffic over an IP Network that has a different line speed and bit error rate.

2) When the FC Fabric or IP Network encounters congestion.

Even when both the FC Fabric or IP network are of comparable speeds, during the course of operation the FC Fabric or the IP Network could encounter congestion due to transient conditions.

The FC Entity uses Fibre Channel mechanisms for flow control at the FC Receiver Portal based on information supplied by the FCIP Entity regarding flow constraints at the Encapsulated Frame Transmitter Portal. The FCIP Entity uses TCP mechanisms for flow control at the Encapsulated Frame Receiver Portal portal based on information supplied by the FC Entity regarding flow constraints at the FC Transmitter Portal.

Coordination of these flow control mechanisms one of which is credit based and the other of which is window based depends on painstaking design that is outside the scope of this specification.

## 8. Security

### 8.1 Threat Models

Using a general purpose, wide-area network such as an IP Network as a substitute for physical cabling introduces some security problems not normally encountered in Fibre Channel Fabrics. FC interconnect cabling typically is protected physically from outside access. Public IP Networks allow hostile parties to impact the security of the transport infrastructure.





The general effect is that the security of the entire FC Fabric is only as good as the security of the entire IP Network through which it tunnels. The following broad classes of attacks are possible:

1) Unauthorized Fibre Channel elements can gain access to resources through normal Fibre Channel Fabric and processes. Although this is a valid threat, securing the Fibre Channel Fabrics is outside the scope of this document. Securing the IP Network is the issue considered in this specification.

2) Unauthorized agents can monitor and manipulate Fibre Channel traffic flowing over physical media used by the IP Network and

under control of the agent.

3) TCP Connections may be hijacked and used to instantiate an invalid FCIP Link between two peer FCIP Entities.

4) Valid and invalid FCIP Encapsulated frames may be injected on the TCP Connections.

5) The payload of an FCIP Encapsulated frame may be altered or transformed in such a way that it preserves the TCP Checksum transform while altering content.

6) Unauthorized agents can masquerade as a valid FCIP Entities and disturb proper operation of the Fibre Channel Fabric.

7) Denial of Service attacks can be mounted by injecting TCP connection requests and other resource exhaustion operations.

The existing IPSec Security Architecture and protocol suite [13] offers protection from these threats. An FCIP Entity MUST implement portions of the IPSec protocol suite as described in this section.

## 8.2 FC Fabric and IP Network Deployment Models

In the context of enabling a secure FCIP tunnel between FC SANs, the following characteristics of the IP Network deployment are useful to note.

1) The FCIP Entities share a peer-to-peer relationship. Therefore, the administration of security policies applies to all FCIP Entities in an equal manner. This varies from a true Client-Server relationship, where there is an inherent difference in how security policies are administered.





2) Policy administration as well as security deployment and configuration are constrained to the set of FCIP Entities, thereby posing less of a requirement on a scalable mechanism. For example, the validation of credentials can be relaxed to the point where deploying a set of pre-shared keys is a viable technique.

3) TCP connections and the IP Network are terminated at the FCIP Entity. The granularity of security implementation is at the level of the FCIP tunnel endpoint (or FCIP Entity), unlike other applications where there is a user-level termination of TCP connections. User-level objects are not controllable by or visible to FCIP Entities. All user-level security related to FCIP is the responsibility of the Fibre Channel standards [7] and outside the scope of this specification.

### 8.3 FCIP Security Components

FCIP Security compliant implementations MUST implement IPSec Protocol Suite based cryptographic authentication and data integrity [13], as well as confidentiality using algorithms and transforms as described in this section. Also, FCIP implementations MUST meet the secure key management requirements of IPSec protocol suite.

#### 8.3.1 IPSec ESP Authentication and Confidentiality

FCIP Entities MUST implement IPSec ESP [15] in Tunnel Mode for providing Data Integrity and Confidentiality. FCIP Entities MAY implement IPSec ESP in Transport Mode, if deployment considerations require use of Transport Mode.

If Confidentiality is not enabled but Data Integrity is enabled, ESP with NULL Encryption [17] MUST be used.

IPSec ESP for message authentication computes a cryptographic hash over the payload that is protected. While IPSec ESP mandates compliant implementations to support certain algorithms for deriving this hash, FCIP implementations:

- MUST implement HMAC with SHA-1 [14]
- SHOULD implement AES in CBC MAC mode with XCBC extensions [draft-pending]

For ESP Confidentiality, FCIP Entities:

- MUST implement 3DES in CBC mode
- SHOULD implement AES in CTR mode [27]
- MUST implement NULL Encryption [17]

When AES is used, the key size SHALL be at least 128-bits and the block size SHALL be at least 128-bits. CTR mode SHALL conform to the Segmented Integer Counter Mode of operation as described in [draft pending] (a possible source for the pending draft is [28]).

### 8.3.2 Key Management

FCIP Entities MUST use the IKE protocol [16] to establish and maintain a Security Association (SA) for use by the two peers. Manual keying for establishing SA is not permitted since it does not provide the necessary elements for rekeying (see section 8.3.3).

IKE Phase 1 establishes a secure, MAC-authenticated channel for communications for use by IKE Phase 2. FCIP Entities MUST support "Main Mode" operation in Phase 1 and MAY support "Aggressive Mode" if identity protection is not required.

FCIP Entities negotiate parameters for SA during IKE Phase 2 only using "Quick Mode". For FCIP Entities engaged in IKE "Quick Mode", there is no requirement for PFS (Perfect Forward Secrecy). FCIP Entities engaged in IKE "Quick Mode" are not required to transmit a Key Exchange (KE) payload.

For a given pair of FCIP Entities, the same IKE Phase 1 negotiation can be used for all Phase 2 negotiations; i.e., all TCP connections that are bundled into the single FCIP Link can share the same Phase 1 results.

Repeated rekeying using "Quick Mode" on the same shared secret will over time, reduce the cryptographic properties of that secret. To overcome this, Phase 1 may be invoked periodically to create a new set of IKE shared secrets and related security parameters.

IKE Phase 1 establishment requires key distribution, and FCIP Entities:

- MUST support pre-shared IKE keys
- MAY support public key encryption
- MAY support signature based authentication

When pre-shared keys are used, FCIP Entities SHALL provide a secure administrative interface for entering these keys. Such mechanisms are outside the scope of this document. Support for IKE Oakley Groups is not required.

For the purposes of establishing a secure FCIP Link, the two participating FCIP Entities consult a Security Policy Database

(SPD). FCIP Entities MUST maintain at least one SPD entry for each FCIP_LEP with which they establish secured TCP Connections, using the Switch WWN of the peer FCIP_LEP as its identifier.   This WWN is transmitted as part of the IKE payload, so that multiple connections between the same FCIP_LEP share the same Phase 1 negotiation.

At the end of successful IKE negotiations both FCIP Entities store the SA parameters in their SA database (SAD). The SAD contains the set of active SA entries, each entry containing Sequence Counter Overflow, Sequence Number Counter, Anti-replay Window and the Lifetime of the SA. FCIP Entities SHALL employ a default SA Lifetime of one hour and a default Anti-replay window of 32 sequence numbers.

When a TCP connection is established between two FCIP_DEs, an SA is created for that connection and is identified in the form of a Security Parameter Index (SPI). Each direction of flow on the TCP connection is associated with a different SA and each FCIP_DE MUST maintain the SPI for its outgoing FCIP Encapsulated Frames.

### 8.3.3 ESP Replay Protection and Rekeying issues

FCIP Entities MUST implement Replay Protection against ESP Sequence Number wrap, as described in [16]. In addition, based on the number of bits in the cipher block size, the validity of the key becomes compromised. In both cases, the SA needs to be reestablished.

FCIP Entities MUST use the results of IKE Phase 1 negotiation for initiating an IKE Phase 2 "Quick Mode" exchange and establish new SAs.

To enable smooth transition of SAs, it is recommended that both FCIP Entities refresh the SPI when sequence number counter reaches 2^31 (i.e., half the sequence number space). It also is recommended that the receiver operate with multiple SPIs for the same TCP connection for a period of 2^31 sequence number packets before aging out an SPI.

When multiple SPIs are active the sending side SHALL use the most recently created SPI.

## 8.4 Secure FCIP Link Operation

### 8.4.1 FCIP Link Initialization Steps

When an FCIP Link is initialized, before any FCIP TCP Connections are established, the local SPD is consulted to determine if IKE Phase 1 has been completed with the FCIP_LEP in the peer FCIP Entity, as identified by the WWN.

If Phase 1 is already completed, IKE Phase 2 proceeds. Otherwise, IKE Phase 1 MUST be completed before IKE Phase 2 can start. Both IKE Phase 1 and Phase 2 transactions use UDP Port 500. If IKE Phase 1 fails, the FCIP Link initialization terminates. Otherwise, the FCIP Link initialization moves to TCP Connection Initialization.

### 8.4.2 TCP Connection Security Associations (SAs)

For a TCP Connection establishment, IKE Phase 2 is employed, resulting in an SA, identified by an SPI. All IP datagrams of the TCP Connection MUST carry an ESP header with a valid SPI and Sequence Number to be accepted as valid by the receiving peer.

An implementation is free to perform several IKE Phase 2 negotiations and cache them in its local SPIs, although entries in such a cache can be flushed per current SA Lifetime settings.

When a TCP Connection is terminated, the SA associated with it MUST be removed from the local SAD.

### 8.4.3 Handling data integrity and confidentiality violations

Upon datagram reception, when the authentication MAC on the ESP payload does not match the stored ESP Authentication data, the receiver MUST drop the datagram, which will trigger TCP retransmission. If many such datagrams are dropped, a receiving FCIP Entity MAY close the connection.

An implementation MAY audit such events as a diagnostic aid.

Confidentiality checks MUST be performed if Confidentiality is enabled.

### 8.4.4 Handling SA parameter mismatches

When SA parameters do not match, the TCP connection may reach a point where no traffic moves, or there are excessive TCP retransmissions. In such a case, either side may close the TCP connection or may choose to reestablish another set of SA parameters.

## 9. Performance

## 9.1 Performance Considerations

Traditionally, the links between FC Fabric components have been characterized by low latency and high throughput. The purpose of FCIP is to replace some of these links with an IP Network, where low latency and high throughput are not as certain. It follows that FCIP





Entities and their counterpart FC Entities probably will be interested in optimal use of the IP Network.

Many options exist for ensuring high throughput and low latency appropriate for the distances involved in an IP Network. For example, a private IP Network might be constructed for the sole use of FCIP Entities. The options that are within the scope of this specification are discussed here.

One option for increasing the probability that FCIP data streams will experience low latency and high throughput is the IP QoS techniques discussed in section 9.2. This option can have value when applied to a single TCP connection. Depending on the sophistication of the FC Entity, further value may be obtained by having multiple TCP connections with differing QoS characteristics.

There are many reasons why an FC Entity might request creation of multiple TCP connections within an FCIP_LEP. These reasons include a desire to provide differentiated service for different TCP data connections between FCIP_LEPs or a preference to separately queue different streams of traffic not having a common in-order delivery requirement.

At the time a new TCP connection is created, the FC Entity SHALL specify to the FCIP Entity the QoS characteristics (including but not limited to IP per-hop-behavior) to be used for the lifetime of that connection. This MAY be achieved by having:

a) only one set of QoS characteristics for all TCP connections;
b) a default set of QoS characteristics that the FCIP Entity applies in the absence of differing instructions from the FC Entity; or
c) a sophisticated mechanism for exchanging QoS requirements information between the FC Entity and FCIP Entity each time a new TCP connection is created.

Once established, the QoS characteristics of a TCP connection SHALL NOT be changed, since this specification provides no mechanism for the FC Entity to control such changes. The mechanism for providing different QoS characteristics in FCIP is the establishment of a different TCP connections and associated FCIP_DEs.

When FCIP is used with a network with a large (bandwidth*delay) product, it is RECOMMENDED that FCIP_LEPs use the TCP mechanisms (window scaling and wrapped sequence protection) for Long Fat Networks (LFNs) as defined in RFC 1323 [10].





## 9.2 IP Quality of Service (QoS) Support

Many methods of providing QoS have been devised or proposed. These include (but are not limited to) the following:

- Multi-Protocol Label Switching (MPLS)
- Differentiated Services Architecture (diffserv) -- RFC 2474 [19], RFC 2475 [20], RFC 2597 [21], and RFC 2598 [22] -- and other forms of per-hop-behavior (PHB)
- Integrated Services, RFC 1633 [11]
- IEEE 802.1p

The purpose of this specification is not to specify any particular form of IP QoS but rather to specify only those issues that must be addressed in order to maximize interoperability between FCIP equipment that has been manufactured by different vendors.

It is RECOMMENDED that some form of preferential QoS be used for FCIP traffic to minimize latency and drop precedence. No particular form of QoS is recommended.

If a PHB IP QoS is implemented, it is RECOMMENDED that it interoperate with diffserv (see RFC 2474 [19], RFC 2475 [20], RFC 2597 [21], and RFC 2598 [22]).

If diffserv/PHB QoS is NOT implemented, the DSCP field for all IP packets SHALL be set to '000000'.

## 10. References

The references in this section were current as of the time this specification was approved. This specification is intended to operate with newer version of the referenced documents and looking for newer reference documents is recommended.

[1] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[3] Fibre Channel Backbone (FC-BB), ANSI NCITS.342:200x, March 5, 2001 (www.t11.org).

[4] Fibre Channel Backbone -2 (FC-BB-2), T11 Project 1466-D, (www.t11.org).





[5] Fibre Channel Switch Fabric -2 (FC-SW-2), ANSI NCITS.355:200x, May 23, 2001 (www.t11.org).

[6] Fibre Channel Framing and Signaling (FC-FS), T11 Project 1331-D, Rev 1.2, February 16, 2001 (www.t11.org).

[7] http://www.t11.org

[8] "Transmission Control Protocol", RFC 793, Sept. 1981.

[9] Braden, R., "Requirements for Internet Hosts -- Communication Layers", RFC 1122, October 1989

[10] Jacobson, V., Braden, R. and Borman, D., "TCP Extensions for High Performance", RFC 1323, May 1992.

[11] R. Braden, et. al., ISI, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994

[12] Mills, D., "Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI", RFC 2030, October 1996.

[13] Kent, S. and Atkinson, R., "Security Architecture for the

Internet Protocol", RFC 2401, Nov. 1998.

[14] Krawczyk, H., Bellare, M., and Canetti, R., "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[15] Kent, S. and Atkinson, R., "IP Encapsulating Security Payload (ESP)", RFC 2406, Nov. 1998.

[16] Harkins, D. and Carrel, D., "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[17] Glenn, R., Kent, S., "The NULL Encryption Algorithm and Its Use With IPsec", RFC 2410, Nov. 1998

[18] Thayer, R., Glenn, R., and Doraswamy, N., "IP Security Document Roadmap", RFC 2411, Nov. 1998.

[19] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and Ipv6 Headers", RFC 2474, December 1998.

[20] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W., "An Architecture for Differentiated Services", RFC 2475, Dec. 1998.





[21] Heinanen, J., Baker, F., Weiss, W., Wroclawski, J., "An Assured Forwarding PHB", RFC 2597, June 1999.

[22] Jacobson, V., Nichols, K., Poduri, K., "An Expedited Forwarding PHB Group", RFC 2598, June 1999.

[23] E.Guttman, C. Perkins, J. Veizades, M. Day. Service Location Protocol, version 2, RFC 2608, July, 1999.

[24] Floyd, et al, "SACK Extension", RFC 2883, July 2000.

[25] Weber, Rajagopal, Travostino, Chau, O'Donnell, Monia Merhar, "FC Frame Encapsulation", draft-ietf-ips-fcencapsulation-__.txt (RFC reference and date to be added during standards action).

[26] Peterson, "Finding FCIP Entities Using SLP", draft-ietf-ips-fcip-slp-___.txt (RFC reference and date to be added during

standards action).

[27] Lipmaa, H., Rogaway, P., and Wagner, D., "Comments to NIST Concerning AES Modes of Operation: CTR-Mode Encryption", NIST Workshop on AES Modes of Operation, http://csrc.nist.gov/encryption/modes/proposedmodes/ctr/ctr-spec.pdf

[28] McGrew, D., "Segmented Integer Counter Mode: Specification and Rationale", NIST Workshop on AES Modes of Operation, http://www.mindspring.com/~dmcgrew/sic-mode.pdf

## 11. Bibliography

The following references may prove informative to readers unfamiliar with Fibre Channel.

Kembel, R., "The Fibre Channel Consultant: A Comprehensive Introduction", Northwest Learning Associates, 1998

## 12. Acknowledgments

Funding for the RFC Editor function is currently provided by the Internet Society.





## 13. Authors' Addresses

Murali Rajagopal
LightSand Communications, Inc.
24411 Ridge Route Dr.
Suite 135
Laguna Hills, CA 92653
USA
Phone: +1 949 837 1733 x101
Email: muralir@lightsand.com

Raj Bhagwat
LightSand Communications, Inc.
24411 Ridge Route Dr.
Suite 135
Laguna Hills, CA 92653
USA
Phone: +1 949 837 1733 x104
Email: rajb@lightsand.com

R. Andy Helland
LightSand Communications, Inc.
375 Los Coches Street
Milpitas, CA 95035
USA
Phone: +1 408 404 3119
Fax: +1 408 941 2166
Email: andyh@lightsand.com

Elizabeth G. Rodriguez
Lucent Technologies
1202 Richardson Drive, Suite 104
Richardson, TX 75080
USA
Phone: +1 972 231 0672
Fax: +1 972 644 5857
Email: egrodriguez@lucent.com

Sriram Rupanagunta
Aarohi Communications
3200 Montelena Drive
San Jose, CA 95135
USA
Phone: +1 408 966 8309
Email: sriramr@aarohi-inc.com

Neil Wanamaker
Akara
10624 Icarus Court
Austin, TX 78726
USA
Phone: +1 512 257 7633
Fax: +1 512 257 7877
Email: nwanamaker@akara.com

Steve Wilson
Brocade Comm. Systems, Inc.
1745 Technology Drive
San Jose, CA. 95110
USA
Phone: +1 408 487 8128
Fax: +1 408 487 8101
email: swilson@brocade.com

Bob Snively
Brocade Comm. Systems, Inc.
1745 Technology Drive
San Jose, CA 95110
USA
Phone: +1 408 487 8135
Email: rsnively@brocade.com

Ralph Weber
ENDL Texas, representing Brocade
Suite 102 PMB 178
18484 Preston Road
Dallas, TX 75252
USA
Phone: +1 214 912 1373
Email: roweber@acm.org





David Peterson
Cisco Systems - SRBU
6450 Wedgwood Road
Maple Grove, MN 55311
USA

Donald R. Fraser
Compaq Computer Corporation
301 Rockrimmon Blvd., Bldg. 5
Colorado Springs, CO 80919
USA

Phone: +1 763 398 1007
Cell: +1 612 802 3299
Email: dap@cisco.com

Phone: +1 719 548 3272
Email: don.fraser@compaq.com

Vi Chau
Gadzoox Networks, Inc.
16241 Laguna Canyon Road
Suite 100
Irvine, CA 92618
USA
Phone: +1 949 789 4639
Fax: +1 949 453 1271
Email: vchau@gadzoox.com

Gaby Hecht
Gadzoox Network, Inc.
16241 Laguna Canyon Road
Suite 100
Irvine, CA 92618
USA
Phone: +1 949 789 4642
Fax: +1 949 453 1271
Email: ghecht@Gadzoox.com

Ken Hirata
Vixel Corporation
15245 Alton Parkway, Suite 100
Irvine, CA 92618
USA
Phone: +1 949 788 6368
Fax: +1 949 753 9500
Email: ken.hirata@vixel.com

Jim Nelson
Vixel Corporation
15245 Alton Parkway, Suite 100
Irvine, CA 92618
USA
Phone: +1 949 450 6159
Fax: +1 949 753 9500
Email: Jim.Nelson@vixel.com

Michael E. O'Donnell
McDATA Corporation
310 Interlocken Parkway
Broomfield, Co. 80021
USA
Phone: +1 303 460 4142
Fax: +1 303 465 4996
Email: modonnell@mcdata.com

Anil Rijhsinghani
McDATA Corporation
5 Brickyard lane
Westboro, MA 01581
USA
Phone: +1 508 870 6593
Email:
anil.rijhsinghani@mcdata.com

Milan J. Merhar
43 Nagog Park
Pirus Networks
Acton, MA 01720
USA
Phone: +1 978 206 9124
Email: Milan@pirus.com

Craig W. Carlson
QLogic Corporation
6321 Bury Drive
Eden Prairie, MN 55346
USA
Phone: +1 952 932 4064
Email: craig.carlson@qlogic.com

Venkat Rangan
Rhapsody Networks Inc.
3450 W. Warren Ave.
Fremont, CA 94538
USA
Phone: +1 510 743 3018
Fax: +1 510 687 0136
Email: venkat@rhapsodynetworks.com

Larwrence J. Lamers
SAN Valley
4611 Park Norton Place
San Jose, CA 95136-2523
USA
Phone: +1 408 626 1285
Email: ljlamers@ieee.org

## 14. Full Copyright Statement



## ANNEX A - Example of synchronization recovery algorithm

The contents of this annex are informative.

Synchronization may be recovered as specified in section 5.6.2.3. An example of an algorithm for searching the bytes delivered to the Encapsulated Frame Receiver Portal for a valid FCIP Frame header is provided in this annex.

This resynchronization uses the principle that a valid FCIP data stream must contain at least one valid header every 2176 bytes (the

maximum length of an encapsulated FC Frame). Although other data patterns containing apparently valid headers may be contained in the stream, the FC CRC or FCIP Frame validity of the data patterns contained in the data stream will always be either interrupted by or resynchronized with the valid FCIP Frame headers.

Consider the case shown in figure 8. A series of short FCIP Frames, perhaps from a trace, are embedded in larger FCIP Frames, say as a result of a trace file being transferred from one disk to another. The headers for the short FCIP Frames are denoted SFH and the long FCIP Frame headers are marked as LFH.

```
+-+--+-+----+-+----+-+----+-+-+-+---+-+---
|L|  |S|    |S|    |S|    |S| |L|   |S|
|F|  |F|    |F|    |F|    |F| |F|   |F|...
|H|  |H|    |H|    |H|    |H| |H|   |H|
+-+--+-+----+-+----+-+----+-+-+-+---+-+---
|                             |
|<---------2176 bytes-------->|
```

Fig. 8  Example of resynchronization data stream

A resynchronization attempt that starts just to the right of an LFH will find several SFH FCIP Frames before discovering that they do not represent the transmitted stream of FCIP Frames. Within 2176 bytes plus or minus, however, the resynchronization attempt will encounter an SFH whose length does not match up with the next SFH because the LFH will fall in the middle of the short FCIP Frame pushing the next header farther out in the byte stream.

Note that the resynchronization algorithm cannot forward any prospective FC Frames to the FC Transmitter Portal because until synchronization is completely established there is no certainty that anything that looked like an FCIP Frame really was one. For example, an SFH might fortuitously contain a length that points exactly to the beginning of an LFH. The LFH would identify the correct beginning of a transmitted FCIP Frame, but that in no way guarantees that the SFH was also a correct FCIP Frame header.

There exist some data streams that cannot be resynchronized by this algorithm. If such a data stream is encountered, the algorithm causes the TCP connection to be closed.

The resynchronization assumes that security and authentication procedures outside the FCIP Entity are protecting the valid data stream from being replaced by an intruding data stream containing valid FCIP data.





The following steps are one example of how an FCIP_DE might resynchronize with the data stream entering the Encapsulated Frame Receiver Portal.

1) Search for candidate and strong headers:

   The data stream entering the Encapsulated Frame Receiver Portal is searched for 12 bytes in a row containing the required values for:
   a) Protocol field,
   b) Version field,
   c) ones complement of the Protocol field,
   d) ones complement of the Version field,
   e) replication of encapsulation word 0 in word 1, and
   f) Reserved field and its ones complement.

   If such a 12-byte grouping is found, the FCIP_DE assumes that it has identified bytes 0-2 of a candidate FCIP encapsulation header.

   All bytes up to and including the candidate header byte are discarded.

   If no candidate header has been found after searching a specified number of bytes greater than some multiple of 2176 (the maximum length of an FCIP Frame), resynchronization has failed and the TCP/IP connection is closed.

   Word 3 of the candidate header contains the Frame Length and Flags fields and their ones complements. If the fields are consistent with their ones complements, the candidate header is considered a strong candidate header. The Frame Length field is used to determine where in byte stream the next strong candidate header should be and processing continues at step 2).

2) Use multiple strong candidate headers to locate a verified candidate header:

The Frame Length in one strong candidate header is used to skip incoming bytes until the expected location of the next strong candidate header is reached. Then the tests described in step 1) are applied to see if another strong candidate header has successfully been located.

All bytes skipped and all bytes in all strong candidate headers processed are discarded.

Strong candidate headers continue to be verified in this way for at least 4352 bytes (twice the maximum length of an FCIP Frame).





If at anytime a verification test fails, processing restarts at step 1 and a retry counter is incremented. If the retry counter exceeds 3 retries, resynchronization has failed and the TCP connection is closed.

After strong candidate headers haves been verified for at least 4352 bytes, the next header identified is a verified candidate header and processing continues at step 3).

Note: If a strong candidate header was part of the data content of an FCIP Frame, the FCIP Frame defined by that or a subsequent strong candidate header will eventually cross an actual header in the byte stream. As a result it will either identify the actual header as a strong candidate header or it will lose synchronization again because of the extra 28 bytes in the length, returning to step 1 as described above.

3) Use multiple strong candidate headers to locate a verified candidate header:

Incoming bytes are skipped and discarded until the next verified candidate header is reached. Each verified candidate header is tested against the full collection of tests listed in section 5.6.2.2 as would normally be the case.

Verified candidate headers continue to be located and tested in this way for a minimum of 4352 bytes (twice the maximum length of an FCIP Frame). If all verified candidate headers encountered are valid, the last verified candidate header is a valid header. At this point the FCIP_DE stops discarding bytes and begins normal FCIP de-encapsulation begins, including for the first

time since synchronization was lost, delivery of FC frames through the FC Transmitter Portal according to normal FCIP rules.

If any verified candidate headers are invalid but meet all the requirements of a strong candidate header, increment the retry counter and return to step 2). If any verified candidate headers are invalid and fail to meet the tests for a strong candidate header, increment the retry counter and return to step 1. If the retry counter exceeds 4 retries, resynchronization has failed and the TCP/IP connection is closed.

A flowchart for this algorithm can be found in figure 9.





```
                         Synchronization is lost
                                  |
                     _____________v_______________
                    |                             |
                    | Search for candidate header |
       +----------->|                             |
       |            |   Found           Not Found |
       |            | (Strong candidate)          |
       |            |_____________________________|
       |                   |              |
       |                   |              + --------->Close TCP/IP
       |             _______v_____________________     Connection
       |            |                             |
       |            |   Enough strong candidate   |
       |      +---->|     headers identified?     |
       |      |     |                             |
       |      |     |     No                Yes   |
       |      |     |        (Verified candidate) |
       |      |     |_____________________________|
       |___________________|                |
       ^      |                             |
       |      |                             |
       |      |      _______________________v_____
       |      |     |                             |
       |      |     | Enough verified candidate   |
```

```
       |      |     |   headers validated?        |
       |      |     |                             |
       |      |     |     No               Yes    |
       |      |     |            (Resynchronized) |
       |      |     |_____________________________|
       |      |            |                |
       |      |      ______v__________      |      Resume
       |      |     |                 |     + ---> Normal
       |      |     | Synchronization |            De-encapsulation
       |      |     |      Lost?      |
       |      |     |                 |
       |      |     | No          Yes |
       |      |     |_________________|
       |      |        |           |
       |      |________|           |
       |___________________________|

       Fig. 9  Flow diagram of simple synchronization example
```





## ANNEX B - Relationship between FCIP and IP over FC (IPFC)

The contents of this annex are informative.

IPFC (RFC 2625) describes the encapsulation of IP packets in FC Frames. It is intended to facilitate IP communication over an FC network.

FCIP describes the encapsulation of FC Frames in TCP segments which in turn are encapsulated inside IP packets for transporting over an IP network. It gives no consideration to the type of FC Frame that is being encapsulated. Therefore, the FC Frame may actually contain an IP packet as described in the IP over FC specification (RFC 2625). In such a case, the data packet would have:

- Data Link Header
- IP Header
- TCP Header
- FCIP Header

- FC Header
- IP Header

Note:   The two IP headers would not be identical to each other. One would have information pertaining to the final destination while the other would have information pertaining to the FCIP Entity.

The two documents focus on different objectives. As mentioned above, implementation of FCIP will lead to IP encapsulation within IP. While perhaps inefficient, this should not lead to issues with IP communication. One caveat: if a Fibre Channel device is encapsulating IP packets in an FC Frame (e.g. an IPFC device), and that device is communicating with a device running IP over a non-FC medium, a second IPFC device may need to act as a gateway between the two networks. This scenario is not specifically addressed by FCIP.

There is nothing in either of the specifications to prevent a single device from implementing both FCIP and IP-over-FC (IPFC), but this is implementation specific, and is beyond the scope of this document.

## ANNEX C - FC Frame Format

The contents of this annex are informative.

All FC Frames have a standard format (see FC-FS [6]) much like LAN's 802.x protocols. However, the exact size of each FC Frame varies depending on the size of the variable fields. The size of the variable field ranges from 0 to 2112-bytes as shown in the FC Frame





Format in figure 10 resulting in the minimum size FC Frame of 36 bytes and the maximum size FC Frame of 2176 bytes. Valid FC Frame lengths are always a multiple of four bytes.

```
+------+--------+-----------+----//-------+------+------+
| SOF  |Frame   |Optional   |  Frame      | CRC  |  EOF |
| (4B) |Header  |Header     | Payload     | (4B) | (4B) |
|      |(24B)   |<----------------------->|      |      |
|      |        | Data Field = (0-2112B)  |      |      |
+------+--------+-----------+----//-------+------+------+
```

Fig. 10  FC Frame Format

## C.1 SOF and EOF Delimiters

On an FC link, Start-of-Frame (SOF) and End-Of-Frame (EOF) are called Ordered Sets and are sent as special words constructed from the 8B/10B comma character (K28.5) followed by three additional 8B/10B data characters making them uniquely identifiable in the data stream.

On an FC link the SOF delimiter serves to identify the beginning of an FC Frame and prepares the receiver for FC Frame reception. The SOF contains information about the FC Frame's Class of Service, position within a sequence, and in some cases, connection status.

The EOF delimiter identifies the end of the FC Frame and the final FC Frame of a sequence. In addition, it serves to force the running disparity to negative. The EOF is used to end the connection in connection-oriented classes of service.

A special EOF delimiter called EOFa (End Of Frame - Abort) is used to terminate a partial FC Frame resulting from a malfunction in a link facility during transmission. Since an FCIP Entity functions like a transmission link with respect to the rest of the FC Fabric, FCIP_DEs may use EOFa in their error recovery procedures.

It is therefore important to preserve the information conveyed by the delimiters across the IP-based network, so that the receiving FCIP Entity can correctly reconstruct the FC Frame in its original SOF and EOF format before forwarding it to its ultimate FC destination on the FC link.

When an FC Frame is encapsulated and sent over a byte-oriented interface, the SOF and EOF delimiters are represented as sequences of four consecutive bytes, which carry the equivalent Class of Service and FC Frame termination information as the FC ordered sets. The representation of SOF and EOF in an encapsulation FC Frame is





described in FC Frame Encapsulation [25].

## C.2 Frame Header

The FC Frame Header is transparent to the FCIP Entity. The FC Frame Header is 24 bytes long and has several fields that are associated with the identification and control of the payload. Current FC

Standards allow up to 3 Optional Header fields [6]:

- Network_Header (16-bytes)
- Association_Header (32-bytes)
- Device_Header (up to 64-bytes).

## C.3 Frame Payload

The FC Frame Payload is transparent to the FCIP Entity. An FC application level payload is called an Information Unit at the FC-4 Level. This is mapped into the FC Frame Payload of the FC Frame. A large Information Unit is segmented using a structure consisting of FC Sequences. Typically, a Sequence consists of more than one FC Frame. FCIP does not maintain any state information regarding the relationship of FC Frames within a FC Sequence.

## C.4 CRC

The FC CRC is 4 bytes long and uses the same 32-bit polynomial used in FDDI and is specified in ANSI X3.139 Fiber Distributed Data Interface. This CRC value is calculated over the entire FC header and the FC payload; it does not include the SOF and EOF delimiters.

Note: When FC Frames are encapsulated into FCIP Frames, the FC Frame CRC is untouched by the FCIP Entity.

# ANNEX D - FC Encapsulation Format

This annex contains a reproduction of the FC Encapsulation Format [25] as it applies to FCIP Frames. The information in this annex was correct as of the time this specification was approved. The information in this annex is informative only.

If there are any differences between the information here and the FC Encapsulation Format specification [25], the FC Encapsulation Format specification takes precedence.

If there are any differences between the information here and the contents of section 5.6.1, then the contents of section 5.6.1 take precedence.

Figure 11 applies the requirements stated in section 5.6.1 and in the FC Encapsulation Frame format resulting in a summary of the FCIP frame format. Where FCIP requires specific values, those values are shown in hexadecimal in parentheses. Detailed requirements for the FCIP usage of the FC Encapsulation Format are in section 5.6.1.

```
W|------------------------------Bit------------------------------|
o|                                                               |
r|3 3 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1                    |
d|1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0|
 +---------------+---------------+---------------+---------------+
0|   Protocol#   |    Version    |  -Protocol#   |   -Version    |
 |    (0x01)     |    (0x01)     |    (0xFE)     |    (0xFE)     |
 +---------------+---------------+---------------+---------------+
1|   Protocol#   |    Version    |  -Protocol#   |   -Version    |
 |    (0x01)     |    (0x01)     |    (0xFE)     |    (0xFE)     |
 +---------------+---------------+---------------+---------------+
2|            Reserved           |           -Reserved           |
 |           (0x00-00)           |           (0xFF-FF)           |
 +-----------+-------------------+-----------+-------------------+
3|   Flags   |   Frame Length    |  -Flags   |   -Frame Length   |
 |  (0x00)   |                   |  (0x3F)   |                   |
 +-----------+-------------------+-----------+-------------------+
4|                     Time Stamp [integer]                      |
 +---------------------------------------------------------------+
5|                     Time Stamp [fraction]                     |
 +---------------------------------------------------------------+
6|                              CRC                              |
 |                        (0x00-00-00-00)                        |
 +---------------+---------------+---------------+---------------+
7|      SOF      |      SOF      |     -SOF      |     -SOF      |
 +---------------+---------------+---------------+---------------+
8|                                                               |
 +-----            FC frame content (see annex C)           -----+
 |                                                               |
 +---------------+---------------+---------------+---------------+
n|      EOF      |      EOF      |     -EOF      |     -EOF      |
 +---------------+---------------+---------------+---------------+
```

Fig. 11  FCIP Frame Format

The names of fields are generally descriptive on their contents and the FC Encapsulation Format specification [25] is referenced for details. Field names preceded by a minus sign are one's complement values of the named field.

ANNEX E - FCIP Requirements on an FC Entity

The contents of this annex are informative for FCIP but might be considered normative on FC-BB-2.

The capabilities that FCIP requires of an FC Entity include:

1) The FC Entity must deliver FC Frames to the correct FCIP Data Engine (in the correct FCIP Link Endpoint).

2) Each FC Frame delivered to an FCIP_DE must be accompanied by a time value synchronized with the clock maintained by the FC Entity at the other end of the FCIP Link (see section 6). If a synchronized time value is not available, a value of zero must accompany the FC Frame.

3) When FC Frames exit FCIP Data Engine(s) via the FC Transmitter Portal(s), the FC Entity should forward them to the FC Fabric. However, before forwarding a FC Frame the FC Entity must compute the end-to-end transit time for the FC Frame using the time value supplied by the FCIP_DE (taken from the FCIP header) and a synchronized time value (see section 6). If the end-to-end transit time exceeds the requirements of the FC Fabric, the FC Entity is responsible for discarding the FC Frame.

4) The only delivery ordering guarantee provided by FCIP is correctly ordered delivery of FC Frames between a pair of FCIP Data Engines. FCIP expects the FC Entity to implement all other FC Frame delivery ordering requirements.

5) The FC Entity may participate in determining allowed TCP connections, TCP connection parameters, quality of service usage, and security usage by modifying interactions with the FCIP Entity that are modelled as a "shared" database in section 7.1.1.

6) The FC Entity may require the FCIP Entity to perform TCP Close requests.

7) The FC Entity may recover from connection failures.

8) The FC Entity must recover from events that the FCIP Entity cannot handle, such as:
   a) loss of synchronization with FCIP Frame headers from the Encapsulated Frame Receiver Portal requiring resetting the TCP connection;

b) recovering from FCIP Frames that are discarded as a result of synchronization problems (see section 5.6.2.2 and section 5.6.2.3);

c) additional examples, TBD

9) The FC Entity must work cooperatively with the FCIP Entity to manage flow control problems in either the IP Network or FC Fabric.

10) The FC Entity may test for failed TCP connections.

11) TBD support for monitoring

Note that the Fibre Channel standards MUST be consulted for a complete understanding of the requirements placed on an FC Entity.

The following table shows the explicit interactions between the FCIP Entity and the FC Entity.

| Reference Section | Condition | Information/Parameter Passed and Direction: FCIP Entity---> | Information/Parameter Passed and Direction: <---FC Entity |
|---|---|---|---|
| 5.6 FCIP Data Engine | FC Frame ready for IP transfer | | Provide FC Frame and time stamp at FC Receiver Portal |
| 5.6 FCIP Data Engine | FCIP Frame received from IP Network | Provide FC Frame and time stamp at FC Transmitter Portal | |
| 5.6.2.2 Errors in FCIP Headers and | FCIP_DE discards bytes delivered through | Inform FC Entity that bytes have been discarded with | |

| | | | | |
|---|---|---|---|---|
| Discarding FCIP Frames | Encapsulated Frame Receiver Portal | reason code | | |
| continued | | | | |





| Reference Section | Condition | Information/Parameter Passed and Direction | |
|---|---|---|---|
| | | FCIP Entity---> | <---FC Entity |
| concluded | | | |
| 7.1.3 Non-Dynamic Creation of a New TCP Connections | New TCP Connection created based on "shared" database information | Inform FC Entity of new or existing FCIP_LEP and new FCIP_DE | |
| 7.1.4 Dynamic Creation of a New TCP Connections | New TCP Connection created based on SLP service advertisement and "shared" database information | Inform FC Entity of new or existing FCIP_LEP and new FCIP_DE | |
| 7.1.5 Processing Incoming TCP Connect Requests | New TCP Connection created based on incoming TCP Connect request and "shared" database information | Inform FC Entity of new or existing FCIP_LEP and new FCIP_DE | |