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

R. Weber
Brocade

M. Rajagopal
LightSand Communications

F. Travostino
Nortel Networks

FC Frame Encapsulation

M. O'Donnell
McDATA

C. Monia
Nishan Systems

M. Merhar
Pirus Networks

Status of this Memo

Abstract

This is the ips (IP Storage) working group draft describing the common encapsulation format and a procedure for the measurement and calculation of frame transit time through the IP network. This specification is intended for use by any IETF protocol that encapsulates Fibre Channel (FC) frames. This draft describes a frame header containing information mandated for encapsulating, transmitting, de-encapsulating, and calculating the transit times of FC frames.

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. Scope

This document describes common mechanisms for the transport of Fibre Channel frames over an IP network, including the encapsulation format and a mechanism for enforcing the Fibre Channel frame lifetime limits.

The organization responsible for the Fibre Channel standards (NCITS Technical Committee T11) has determined that some functions and modes of operation are not interoperable to the degree required by the IETF. This draft includes applicable T11 interoperability determinations in the form of restrictions on the use of this encapsulation mechanism.

Use of these mechanisms in a protocol requires an additional document to specify the protocol-specific functionality and appropriate security considerations. Because security considerations for this encapsulation depend on how it is used by protocols, they are taken up in protocol-specific documents.

2. Encapsulation Concepts

The smallest unit of data transmission and routing in Fibre Channel (FC) is the frame. FC frames include a Start Of Frame (SOF), End Of Frame (EOF), and the contents of the Fibre Channel frame. The Fibre Channel frame has a CRC that provides error detection for the contents of the frame. FC frames have several possible lengths. To facilitate transporting FC frames over TCP the native FC frame needs to be contained in (encapsulated in) a slightly larger structure as shown in figure 1.

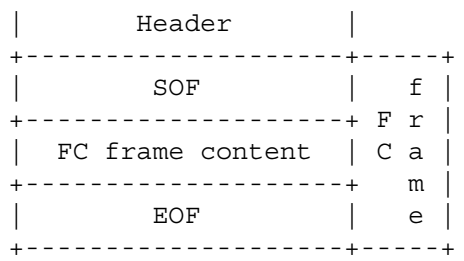

Fig. 1 - FC frame Encapsulation

The format and content of an FC frame is described in the FC standards (e.g., FC-FS [3], FC-SW-2 [4], and FC-PI [5]). Of importance to this encapsulation is the FC requirement that all frames SHALL contain a CRC for detection of transmission errors.

3. The FC Encapsulation Header

3.1 FC Encapsulation Header Format

Figure 2 shows the format of the required FC Encapsulation Header.

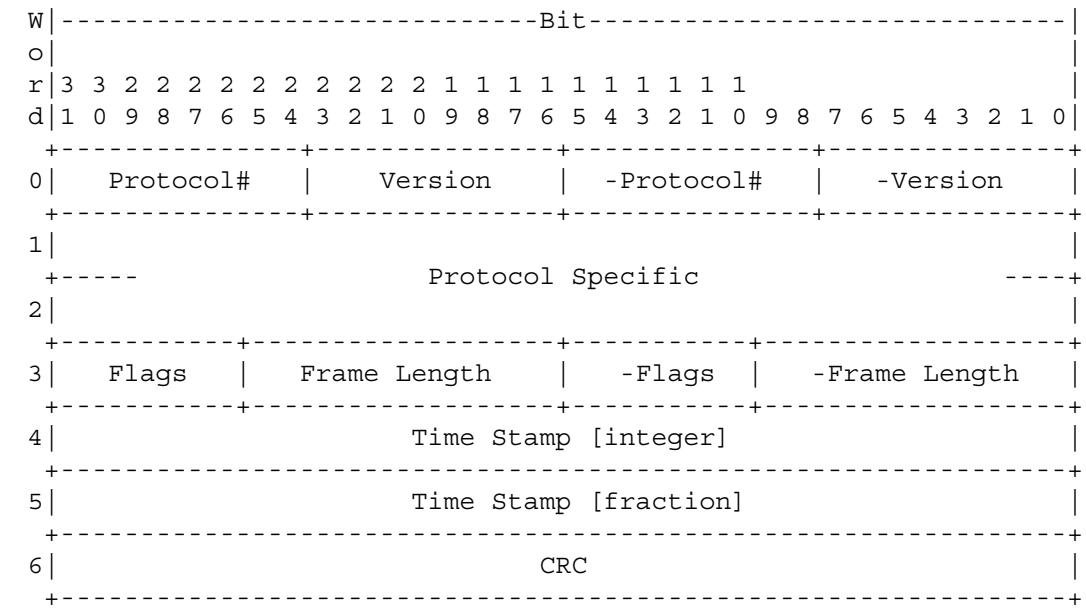

Fig. 2 - FC Encapsulation Header Format

The fields in the FC Encapsulation Header are defined as follows.

Protocol (bits 31-24 in word 0): The Protocol# field SHALL contain a number that indicates which protocol is employing the FC Encapsulation. The values in the Protocol# field are assigned by IANA [8]. The following values are known to be in use:

- FCIP -- TO BE ASSIGNED by IANA
- iFCP -- TO BE ASSIGNED by IANA

Version (bits 23-16 in word 0): The Version field SHALL contain 0x1 to indicate that this version of the FC Encapsulation is being used. All other values are reserved for future versions of the FC Encapsulation.

-Protocol# (bits 15-8 in word 0): The -Protocol# field contains the ones complement of the contents of the Protocol# field. FC Encapsulation receivers may compare the Protocol# and -Protocol# fields as an additional verification that an FC Encapsulation Header is being processed.

-Version (bits 7-0 in word 0): The -Version field contains the ones complement of the contents of the Version field. FC Encapsulation receivers may compare the Version and -Version fields as an additional verification that an FC Encapsulation Header is being processed.

Protocol Specific (words 1 and 2): The usage of these words differs based on the contents of the Protocol# field, i.e., the usage of this word is defined by the protocol that is employing this encapsulation.

Flags (bits 31-26 in word 3): The Flags bits provide information about the usage of the FC Encapsulation Header as shown in figure 3.

Note: Implementers are advised to consult the specifications of protocols that use this header to determine how each individual protocol uses the bits in the Flags field.

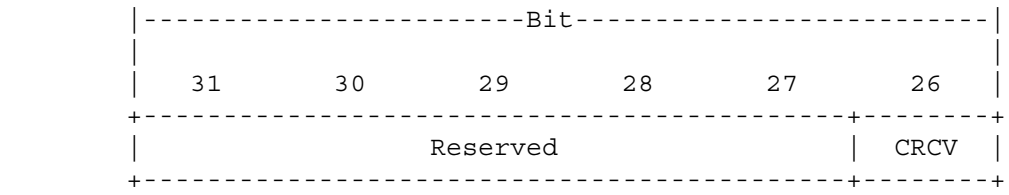

Fig. 3 - Flags Field Format

Reserved (Flags, bits 31-27 in word 3): These bits are reserved for use by future versions of the FC Encapsulation and SHALL be set to zero on send. Protocols employing this encapsulation MAY require checking for zero on receive, however doing so has the potential to create incompatibilities with future versions of this encapsulation. Changes in the usage of the Reserved Flags bits MUST be identified by changes in the contents of the Version field. Protocols employing this encapsulation MUST NOT make use of the Reserved Flags bits in any fashion other than that described here.

CRCV (CRC Valid Flag, bit 26 in word 3): A CRCV bit value of one indicates that the contents of the CRC field are valid. A CRCV bit value of zero indicates that CRC are invalid. Some protocols may always check the CRC without regard for the state of this bit. The value of the CRCV bit SHALL be constant for all FC Encapsulation Headers sent on a given TCP connection.

Frame Length (bits 25-16 in word 3): The Frame Length field contains the length of the entire FC Encapsulated frame including the FC Encapsulation Header and the FC frame (including SOF and EOF words).

This length is based on a unit of 32-bit words. All FC frames are 32-bit-word-aligned and the FC Encapsulation Header SHALL always be word-aligned; therefore a 32-bit word length is acceptable.

-Flags (bits 15-10 in word 3): The -Flags field contains the ones complement of the contents of the Flags field. FC Encapsulation receivers may compare the Flags and -Flags fields as an additional verification that an FC Encapsulation Header is being processed.

-Frame Length (bits 9-0 in word 3): The -Frame Length field contains the ones complement of the contents of the Frame Length field. FC Encapsulation receivers may compare the Frame Length and -Frame Length fields as an additional verification that an FC Encapsulation Header is being processed.

Time Stamp [integer] and Time Stamp [fraction] (words 4 and 5): The two Time Stamp fields contain time at which the FC Encapsulated frame was sent as known to the sender. The format of integer and fraction Time Stamp word values is specified in Simple Network Time Protocol (SNTP) Version 4 [9]. The contents of the Time Stamp [integer] and Time Stamp [fraction] words SHALL be set as described in section 4.

CRC (word 6): When the CRCV Flag bit is zero, the CRC field SHALL contain zero. When the CRCV Flag bit is one, the CRC field SHALL contain a CRC for words 0 to 5 of the FC Encapsulation Header computed using the polynomial, initial value, and bit order defined for Fibre Channel in FC-FS [3]. Using this algorithm, the bit order of the resulting CRC corresponds to that of FC-1 layer. The CRC transmitted over the IP network shall correspond to the equivalent value converted to FC-2 format as specified in FC-FS.

3.2 FC Encapsulation Header Validation

Two mechanisms are provided for validating an FC Encapsulation Header:

- Redundancy based
- CRC based

The two mechanisms address the needs of two different design and operating environments.

3.2.1 Redundancy based FC Encapsulation Header validation

Redundancy based validation of an FC Encapsulation Header relies on duplicated and one's complemented fields in the header.

Redundancy based header validation can be built from simple logic (e.g., XORs and comparisons). Header validation based on redundancy also is a step wise process in that the first word is validated, then the second, then the third and so on. A decision that a candidate header is not valid may be reached before the complete header is available.

3.2.2 CRC based FC Encapsulation Header validation

CRC based validation of an FC Encapsulation Header relies on a CRC located in the last word of the header.

CRC based header validation employs a straight forward algorithm (e.g., compute the CRC for all bytes preceding the CRC word and compare the results to the CRC word's contents). The number of comparisons required to perform CRC validation is exactly one and the method for computing the CRC is well known with proven implementations.

4. Measuring Fibre Channel frame transit time

To comply with FC-FS [3], an FC Fabric must specify and limit the lifetime of a frame. In an FC Fabric comprised of TCP-connected elements, one component of the frame's lifetime is the time required to traverse the TCP connection. To ensure that the total frame lifetime remains within the limits required by the FC Fabric, the encapsulation described in this specification contains provisions for recording the departure time of an encapsulated frame injected into a TCP connection. If the encapsulated frame originator and recipient have access to aligned and synchronized time bases, the transit time through the IP network can then be computed.

When originating an encapsulated frame, an entity that does not support transit time calculation SHALL always set the Time Stamp [integer] and Time Stamp [fraction] fields to zero. When receiving an encapsulated frame, an entity that does not support transit time calculation SHALL ignore the contents of the Time Stamp words. The protocol SHALL specify whether or not implementation support is required.

Encapsulating and de-encapsulating entities that support this feature MUST have access to:

a) An internal time base having the stability and resolution necessary to comply with the requirements of the protocol specification; and

b) A time base that is synchronized and aligned with the time base of other entities to which encapsulated frames may be sent or received. The protocol specification MUST describe the synchronization and alignment procedure.

With respect to its ability to measure and set transit time for encapsulated frames exchanged with another device, an entity is either in the Synchronized or Unsynchronized state. An entity is in the Unsynchronized state upon power-up and transitions to the Synchronized state once it has aligned its time base in accordance with the applicable protocol specification.

An entity MUST return to the Unsynchronized state if it is unable to maintain synchronization of its time base as required by the protocol specification.

The policy for processing frames while in the Unsynchronized state SHALL be defined by the protocol specification, including whether or not the entity may continue to send and receive frames from the IP network.

If processing frames in the Unsynchronized state is permitted by the protocol specification, the entity SHALL:

a) When de-encapsulating a frame, ignore the Time Stamp words; and

b) When encapsulating a frame set the Time Stamp [integer] and Time Stamp [fraction] words to zero.

When encapsulating a frame, an entity in the Synchronized state SHALL record the value of the time base in the Time Stamp [integer] and Time Stamp [fraction] words in the encapsulation header.

When de-encapsulating a frame, an entity in the Synchronized state SHALL:

a) Test the Time Stamp words to determine if they contain a time as specified in [9]. If the time stamp is valid, the receiving entity SHALL compute the transit time by calculating the difference between its time base and the departure time recorded in the frame header. The receiving entity SHALL process the calculated transit time and the de-encapsulated frame in accordance with the applicable protocol specification; or

b) If both Time Stamp words have a value of zero, the receiving entity SHALL process the de-encapsulated frame without computing the transit time. The disposition of the frame and any other actions by the recipient SHALL be defined by the protocol specification.

5. The FC frame

5.1 FC frame content

Figure 4 shows the structure of a general FC-2 frame format.

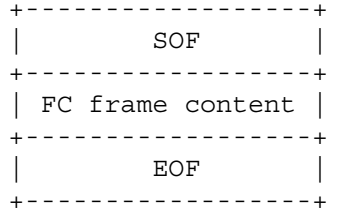

Fig. 4 - General FC-2 Frame Format

As shown in figure 4, the FC frame content is defined as the data between the EOF and SOF delimiters (including the FC CRC) after conversion from FC-1 to FC-2 format as specified by FC-FS [3].

5.2 Bit and Byte Ordering

The Encapsulation Header, SOF, FC frame content (see section 5.1), and EOF are mapped to TCP using the big endian byte ordering, which corresponds to the standard network byte order or canonical form [10].

5.3 FC SOF and EOF

The FC frame content is composed of 8-bit bytes that can be translated directly for transmission over TCP. The FC SOF and EOF [3] require 8b/10b special characters that cannot be translated directly to 8-bit bytes, encoded values are required.

For this reason, the encapsulated FC frame SHALL have the format shown in figure 5. The redundancy of the SOF/EOF representation in the encapsulation format results from concerns that the information be protected from transmission errors.

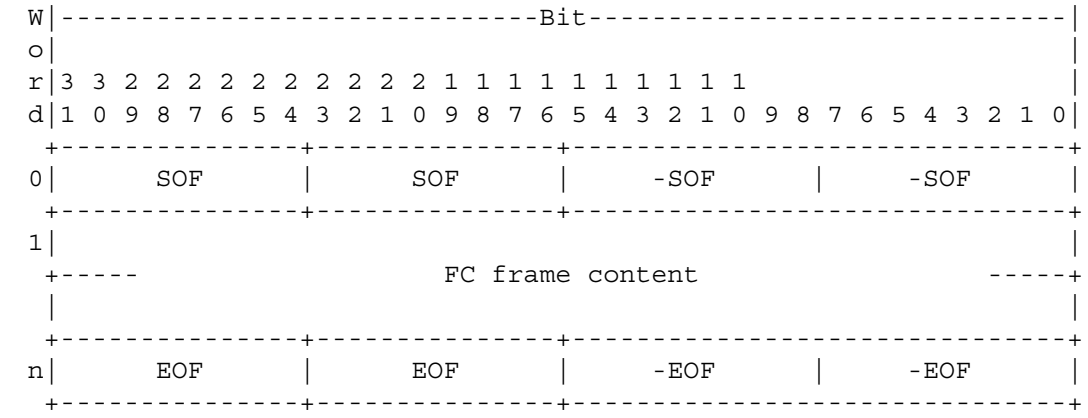

Fig. 5 - FC Frame Encapsulation Format

SOF (bits 31-24 and bits 23-16 in word 0): The SOF fields contain the encoded SOF value selected from table 1.

FC SOF	SOF Code
SOFf	0x28
SOFi2	0x2D
SOFn2	0x35
SOFi3	0x2E
SOFn3	0x36
SOFi4	0x29
SOFn4	0x31
SOFc4	0x39

Table 1 Translation of FC SOF values to SOF field contents

-SOF (bits 15-8 and 7-0 in word 0): The -SOF fields contain the one's complement of the value in the SOF fields.

EOF (bits 31-24 and 23-16 in word n): The EOF fields contain the encoded EOF value selected from table 2.

FC EOF	EOF Code		FC EOF	EOF Code
EOFn	0x41		EOFdt	0x46
EOFt	0x42		EOFdti	0x4E
EOFni	0x49		EOFrt	0x44
EOFa	0x50		EOFrti	0x4F

Table 2 Translation of FC EOF values to EOF field contents

-EOF (bits 15-8 and 7-0 in word n): The -EOF fields contain the one's complement of the value in the EOF fields.

Note: FC-BB-2 [6] lists SOF and EOF codes not shown in table 1 and table 2 (e.g., SOFi1 and SOFn1). However, FC-MI [7] identifies these codes as not interoperable, so they are not listed in this specification.

6. Security

This document describes the encapsulation format only. Actual use of this format in a protocol requires an additional document to specify the protocol functionality and appropriate security considerations. Because security considerations for this encapsulation depend on how it is used by protocols, they SHALL be described in protocol-specific documents.

7. Normative References

[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 Framing and Signaling (FC-FS), T11 Project 1331-D, (http://www.t11.org/t11/docreg.nsf/ldl/fc-fs).

[4] Fibre Channel Switch Fabric -2 (FC-SW-2), ANSI NCITS.355:200x, May 23, 2001 (http://www.t11.org/t11/docreg.nsf/ldl/fc-sw-2).

[5] Fibre Channel Physical Interfaces (FC-PI), ANSI NCITS.352:200x, August 18, 2000.

[6] Fibre Channel Backbone -2 (FC-BB-2), T11 Project 1466-D, (http://www.t11.org/t11/docreg.nsf/ldl/fc-bb-2).

[7] Fibre Channel Methodologies for Interconnects (FC-MI), T11 Project 1377-D, (http://www.t11.org/t11/docreg.nsf/ldl/fc-mi).

[8] Reynolds, J. and Postel, J., "Assigned Numbers", RFC 1700, October, 1994.

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

[10] Narten, T. and C. Burton, "A Caution on The Canonical Ordering of Link-Layer Addresses", RFC 2469, December 1998.

8. Authors' Addresses

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

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

Franco Travostino
Technology Center
Nortel Networks, Inc.
600 Technology Park
Billerica, MA 01821
USA
Phone: +1 978 288 7708
Email: travos@nortelnetworks.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

Charles Monia
Nishan Systems
3850 North First Street
San Jose, CA 95134
USA
Phone: +1 408 519 3986
Email: cmonia@nishansystems.com

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

9. Acknowledgements

The authors express their appreciation to Mr. Vi Chau (vchau1@cox.net) for his contributions to the design team that developed this document. Mr. Chau is no long working in this technology.

10. Full Copyright Statement

ANNEX A - Protocol Requirements

This annex lists the requirements placed on the protocols that employ this encapsulation. The requirements listed here are suggested or described elsewhere in this document, but their collection in this annex serves to assist protocol authors in meeting all obligations placed upon them.

Protocol Specific Data

Protocols employing this encapsulation SHALL:

- specify the IANA assigned number used in the Protocol# field
- specify the contents of the Protocol Specific field

CRC

Protocols employing this encapsulation SHALL either:

1) Require a valid CRC to be sent and the CRCV Flag bit to be sent as one, or
2) Require the CRC field to be sent as zero and the CRCV Flag bit to be sent as zero.

Protocols employing this encapsulation SHALL define the procedures and policies necessary for the detection of over age frames. The items to be specified and the choices available to a specification are as follows:

a) The protocol requirements for measuring transit times. The protocol MAY allow implementation of transit time measurement to be optional.

b) The requirements or guidelines for stability and resolution of the entity's time base.

c) The procedure for synchronizing an entity's time base, including the criteria for entering the Synchronized and Unsynchronized states.

d) The processing of frame traffic while in the Unsynchronized state.

   The specification MAY allow an entity in the Unsynchronized state to continue processing frame traffic.

e) The procedure to be followed when frames are received that do not have a valid time stamp.

   The specification MAY allow such frames to be accepted by the entity.

f) Requirements for setting and testing the transit time limit and the procedure to be followed when a received frame is discarded due to its transit time exceeding the limit.

ANNEX B - IANA Considerations

The Protocol# (Protocol Number, bits 31-24 in word 0 of the Encapsulation Header) field is an identifier number used to distinguish between the protocols that employ this encapsulation. Values used in the Protocol# field are to be assigned from a new, separate registry that is maintained by IANA in accordance with RFC 1700 [8].

All values in the Protocol# field are to be registered with and assigned by IANA with the following exceptions.

- Protocol# value 0 should not be assigned until after all other values have been assigned.

- Protocol# values 240-255 inclusive must be set aside for private use amongst cooperating systems.

Standards action on this RFC should be accompanied by IANA assignment of the following two Protocol# values:

- Protocol# value 1 assigned to the FCIP (Fibre Channel Over TCP/IP) protocol being developed in draft-ietf-ips-fcovertcpip-__.txt.

- Protocol# value 2 assigned to the iFCP (A Protocol for Internet Fibre Channel Storage Networking) protocol being developed in draft-ietf-ips-ifcp-__.txt.

Requests for assignments of Protocol# values must be accompanied by an RFC which describes how this encapsulation is employed. If the RFC is not on the standards-track (i.e., it is an informational or experimental RFC), it must be explicitly reviewed and approved by the IESG before the RFC is published and Protocol# value is assigned. It is requested that the ips working group chairs or the Transport Services area directors be notified when any new Protocol# value assignment is requested.