iFCP Encapsulation Requirements and Guidelines
draft-monia-ips-ifcpencap-00.txt

Charles Monia

Senior Technology Consultant

Nishan Systems

cmonia@nishansystems.com

iFCP Encapsulation Assumptions

- TCP/IP layering is preserved.
  - All encapsulation and TCP framing operations occur in higher layers

- Framing assists in the TCP/IP stream are separate from FC frame encapsulation.
  - TCP/IP framing assists
    - Data added to facilitate PDU recovery before the PDU is injected into the TCP/IP stream
    - E.g., Word-stuffing is a TCP framing assist. It is not part of the encapsulation.
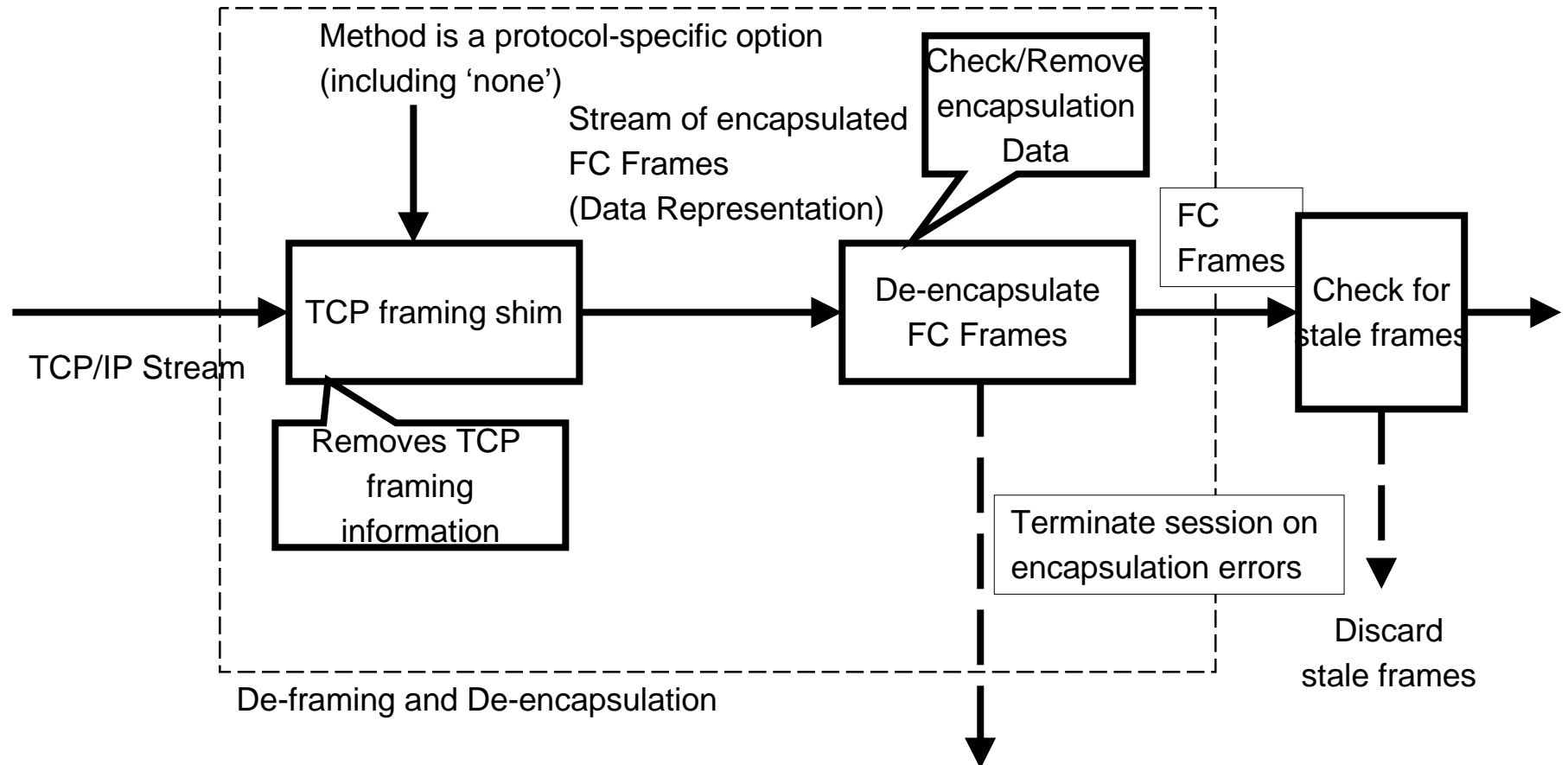 - Should be done by 'shim layer'

iFCP Encapsulation Error Detection

- iFCP only checks for errors in the encapsulation data referenced during iFCP de-encapsulation.
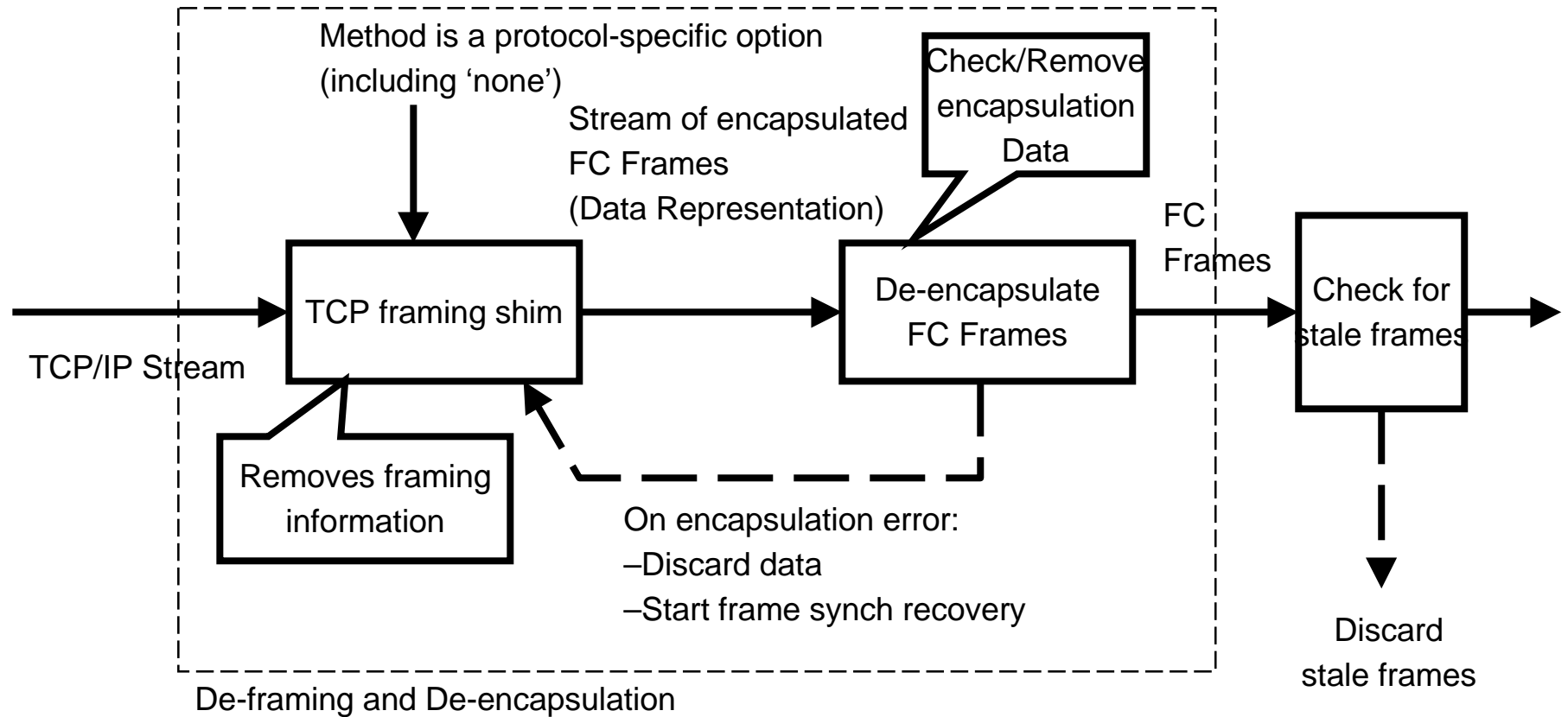- FC frame 'digest' (ie. CRC) generated and checked by FC layer only

Encapsulation Error Handling

- ## iFCP terminates TCP/IP connection
  - Rationale:
    - 'Cross section' of data checked during de-encapsulation is small compared to average size of FC frame payload
    - Loss of a TCP/IP connection affects only one N_PORT-to-N_PORT session.

- ## FCIP discards frames and continues the session.
  - Rationale:
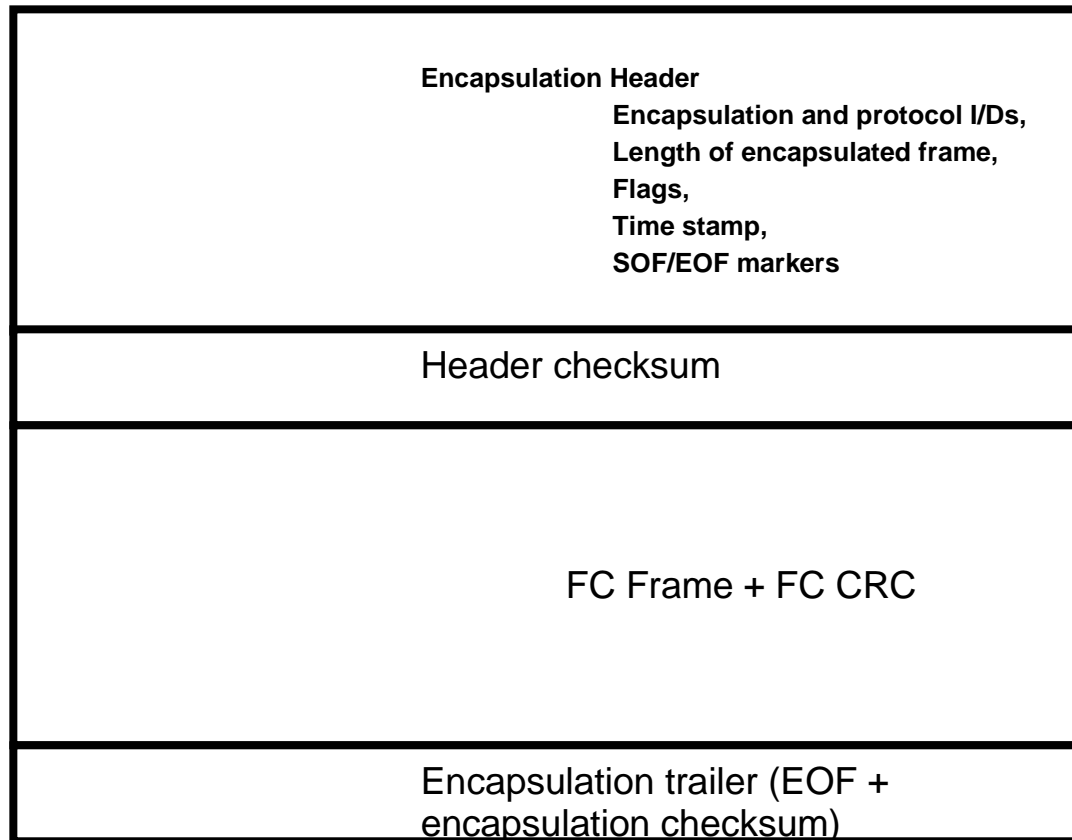   - Aborting a tunnel session may cause fabric-wide disruption

iFCP De-Encapsulation Model

FCIP De-Encapsulation Model

Method is a protocol-specific option
(including 'none')

Stream of encapsulated
FC Frames
(Data Representation)

Check/Remove
encapsulation
Data

FC
Frames

TCP framing shim

De-encapsulate
FC Frames

Check for
stale frames

TCP/IP Stream

Removes framing
information

On encapsulation error:
–Discard data
–Start frame synch recovery

Discard
stale frames

De-framing and De-encapsulation

Template for a Common Data Representation

Encapsulation Header
      Encapsulation and protocol I/Ds,
      Length of encapsulated frame,
      Flags,
      Time stamp,
      SOF/EOF markers

Header checksum
FC Frame + FC CRC
Encapsulation trailer (EOF + encapsulation checksum)

NB: TCP Framing data (e.g., word stuffing), if any, is removed by the framing shim.

iFCP Requirements and Guidelines

- MUST
  - Header must have fixed length component with contiguous digest
    - Allows pipelining of frame processing with frame reception
  - Place all encapsulation data required by iFCP in fixed part of header, including copy of FC EOF encoding.
    - Can be validated with one header digest check
    - Eof delimiter can be replicated at the end of the frame.
    - Checksum at end of frame is OK, but iFCP won't validate it.
  - Allocate space in fixed part of header for protocol-specific flags field.
    - Quick way to detect special frames, such as augmented ELSs.
    - 16 bits is enough

- Issue
  - TCP-style checksums should not be used.
    - Should consider fletcher instead.

iFCP Requirements, Guidelines (con't)

- ## MUST NOT
  - Require all 'client' protocols to implement framing hooks (e.g., word-stuffing)
    - Should be a protocol-specific option
  - Require use of CRC in any digest added by the encapsulation process
    - CRC is overkill for small headers

- ## Should
  - Augment header fields with additional patterns that can be checked for consistency with low overhead and cost
    - Since iFCP will perform an exhaustive consistency check of all header fields on each frame, the overhead for such checks must be low.