CTF-capable (E)ISS

Support for CTF-capable forwarding process functions

Christophe Mangin

2023/11/16

MITSUBISHI ELECTRIC R&D CENTRE EUROPE
introduction
(E)ISS primitives and their parameters

- Reception path
  - From the MAC client interface up to the EISS

\[ P : \text{set of parameters} \]
Store-and-Forward MAC client interface behavior

- 802.3 MAC as an example
  - Rx operation
    - The PHY layer delineates the frame thanks to physical signalling indicating the presence of successive bit belonging to a frame (RxValid active)
    - Successively received bits of a frame are accumulated in a structure
    - Upon reception of the last bit of a frame (when RxValid transitions to inactive), a set of operations are performed
      - Discarding of the frame if it is too short (the only reason for discard)
      - Identification of the different parameters/fields of the frame structure
        - Addresses, type/length, msdu, FCS
        - Determination of the frame reception status
          - Frame too long, octets misalignment, FCS validity, ...
      - If the frame is not discarded, the MA_DATA.indication primitive is sent to the MAC client, including the following parameters:
        - Received frame structure
        - Frame reception status
        - The size of the structure is implicitly known
      - The M_DATA.indication signals the availability of received frame structure to the MAC client

MA_DATA.indication

{ destination_address, source_address, mac_service_data_unit, frame-check-sequence, reception_status }

RxValid envelope

M_DATA.indication

{ destination_address, source_address, mac_service_data_unit, frame-check-sequence, reception_status }

Frame data structure

dest address

source address

mac service data unit

FCS

Frame data structure
Properties of the (E)ISS primitives

• The ISS primitives are transformations of the MAC client interface primitives

• The EISS primitives are transformations of the ISS primitives

• If the MAC is Store-and-Forward so is the (E)ISS
CTF-capable (E)ISS modelling proposal
MAC Relay Entity: the Store-and-Forward case

- MAC relay entity’s forwarding process in Store-and-Forward bridges
  - Performed by a series of functions taking actions on a given frame (totally) received on a given port
    - Function invoked upon reception of the (E)M_UNITDATA.indication primitive
    - Function decision: forward or discard
  - Function’s decision = f(p_F, t, L)
    - p_F: the subset of P, provided by the (E)M_UNITDATA.indication primitive, that is necessary to establish the function decision
    - t: the time the frame is received at
    - L: the length of the frame, L = f(P)

Forwarding process functions (802.1Q-2022)
MAC Relay Entity: the Cut-Through case

- MAC relay entity’s forwarding process in Cut-Through bridges
  - Performed by the same series of functions (or a subset thereof)
    - Some functions can possibly not support CTF
  - Cut-Through:
    - Invocation of each function occurs at a time when the frame is not totally received
    - Function’s action = result of an early/interim decision
      - Missing or incomplete function input at the time of function invocation
      - Function’s action can be corrected once the missing or incomplete function’s input is available or completed.
  - Function’s input that is incomplete until the end of the frame reception
    - L: does not prevent from taking some decisions
    - msdu: partial info is enough to take a decision
    - FCS: considered valid until it can be checked
Principles for a CTF-capable (E)ISS

• The CTF-capable (E)ISS is compatible with the legacy (E)ISS (802.1AC and 802.1Q)
  • The MAC relay entity of a CTF-capable bridge can support legacy (S&F) (E)ISS providers
  • The MAC relay entity of a legacy (S&F) bridge can support CTF-capable (E)ISS providers

• The primitives of the CTF-capable (E)ISS are designed so that the MAC relay entity functions do not have to perform functions that are « normally » performed in other layers
  • Invalid frame format detection (length, alignment, …)
  • MAC parameter delineation
  • FCS checking

• The set of CTF-capable (E)ISS primitives is a superset of the set of legacy (E)ISS primitives

• « Fall-back to S&F » just consists in:
  • Stopping current CTF processing of the frame being received and « wait for » an indication primitive marking the end of the frame reception to resume the execution of the forwarding functions
CTF-capable (E)ISS primitives: frame reception

- (E)M_JUST-ENOUGH_UNITDATA.indication
  - Generated when the greatest common set of frame parameters required by any of the MAC Relay Entity functions, supported for a given port, can be made available over the (E)ISS interface.
    - e.g. Stream filtering is implemented and IP Stream identification is one of the Stream identification functions used to determine the Stream handle
      - Required frame parameters: destination address, vlanid, set of initial msdu octets used by the IP Stream identification function
  - Triggers continuous provision, over the (E)ISS, of the data of the remaining incomplete parameters along the reception of the frame by the lower layers
  - L*: partial length of the frame at the time (E)M_JUST-ENOUGH_UNITDATA.indication is generated
    - L* increases along the continuous provision of the remaining part of the frame
  - FCS check is unavailable at the time (E)M_JUST-ENOUGH_UNITDATA.indication is generated

- (E)M_INVALID_UNITDATA.indication
  - Only generated if an (E)M_JUST-ENOUGH_UNITDATA.indication was previously generated
  - Generated at an instant of time corresponding to the end of the frame reception by the lower layers
    - i.e. at the instant of time the (E)M_UNITDATA.indication would be generated
  - 1 parameter: reason
    - Runt frame, invalid FCS, ...
2 CTF-capable (E)ISS primitives: frame reception

- **(E)M_UNITDATA.indication**
  - Invoked at the end of an error-free frame reception
  - When generated after an (E)M_JUST-ENOUGH_UNITDATA.indication
    - Validates the frame
    - Allows determination of L
    - Determines the time of frame reception completion
  - When not preceded by an (E)M_JUST-ENOUGH_UNITDATA.indication
    - the (E)ISS only supports Store-and-Forward

- **(E)M_UNITDATA.indication and (E)M_INVALID_UNITDATA.indication are exclusive**
  - One or the other signals the end of the frame over the CTF-capable (E)ISS
CTF-capable (E)ISS primitives: frame transmission

- **(E)M_JUST-ENOUGH_UNITDATA.request**
  - Generated when partial data of a Cut-Through-forwarded frame is available in a queue selected for transmission
    - Typically the partial set of parameters that are passed with the (E)M_JUST-ENOUGH_UNITDATA.indication
  - The remaining (parts of the) parameters are progressively provided over the (E)ISS when they are made available by the MAC relay entity
    - The queue selected for transmission is continuously filled up by the CTF-capable MAC relay entity’s frame queueing function

- **(E)M_DISCARD.request**
  - Generated when:
    - An (E)M_INVALID_UNITDATA.indication was received
    - A frame discard condition is detected by one of the MAC relay entity’s functions
      - Can be generated before the end of the frame reception
  - 1 parameter: reason
    - Runt frame, invalid FCS, discard(relaying function), ...
  - Indicates that actions must be taken by the lower layers to:
    - Discard the frame
    - Signal the frame as invalid to downstream network elements
CTF-capable (E)ISS primitives: frame transmission

- (E)M_UNITDATA.request
  - Invoked at the end of an error-free frame transmission
  - When generated after an (E)M_JUST-ENOUGH_UNITDATA.indication
    - Signals the end of the continuous provision of data over the (E)ISS
    - Validates the frame
  - When not preceded by an (E)M_JUST-ENOUGH_UNITDATA.request
    - Signals a (whole) frame transmission to the lower layers supporting Store-and-Forward only

- (E)M_UNITDATA.request and (E)M_DISCARD.request are exclusive
  - One or the other signals the end of the frame over the CTF-capable (E)ISS
Examples
CTF-(E)ISS primitives timeline (reception)

• Valid frame reception

  EM_JUST-ENOUGH_UNITDATA.indication → CTF-EISS → EM_UNITDATA.indication
  M_JUST-ENOUGH_UNITDATA.indication → CTF-ISS → M_UNITDATA.indication

• Invalid frame reception

  EM_JUST-ENOUGH_UNITDATA.indication → CTF-EISS → EM_INVALID_UNITDATA.indication
  M_JUST-ENOUGH_UNITDATA.indication → CTF-ISS → M_INVALID_UNITDATA.indication
CTF-(E)ISS primitives timeline (transmission)

- **Valid frame transmission**
  - EM_JUST-ENOUGH_UNITDATA.request
  - CTF-EISS
  - EM_UNITDATA.request
  - Frame transmission

- **Invalid frame transmission**
  - EM_JUST-ENOUGH_UNITDATA.request
  - CTF-EISS
  - EM_DISCARD.request
  - Frame transmission
  - M_JUST-ENOUGH_UNITDATA.request
  - CTF-ISS
  - M_UNITDATA.request
  - M_DISCARD.request
  - Frame transmission

The S&F lower-layer entity decides what to do (signal/mark/truncate?) with the invalid frame.
CTF-(E)ISS compatibility with (E)ISS (reception)

- Valid frame reception
  - **EM_JUST-ENOUGH_UNITDATA.indication**
  - **M_JUST-ENOUGH_UNITDATA.indication**

- Invalid frame reception
  - **EM_JUST-ENOUGH_UNITDATA.indication**
  - **M_JUST-ENOUGH_UNITDATA.indication**
**CTF-(E)ISS compatibility with (E)ISS (transmission)**

- **Valid frame transmission**

  - Not relayed by the (S&F) ISS

  - Potential CTF-frame transmission

  - CTF-EISS

  - $\text{EM\_UNITDATA\_request}$
  - $\text{EM\_DISCARD\_request}$
  - $\text{M\_UNITDATA\_request}$
  - $\text{M\_DISCARD\_request}$

- **Invalid frame transmission**

  - Not relayed by the (S&F) ISS

  - Potential CTF-frame transmission

  - CTF-ISS

  - $\text{EM\_UNITDATA\_request}$
  - $\text{EM\_DISCARD\_request}$
  - $\text{M\_UNITDATA\_request}$
  - $\text{M\_DISCARD\_request}$

---

** Examples | 18**
Any questions?