ERICSSON

# ECN SUPPORT IN QUIC

draft-johansson-quic-ecn-03

Ingemar Johansson

# INTRO

- ECN = Explicit Congestion Notification
- Makes it possible for congested nodes to mark instead of discard packets
- Look for instance in RFC3168 for more info
- ECN is a key component in L4S
  draft-briscoe-tsvwg-l4s-arch, draft-ietf-tsvwg-ecn-experimentation
- Objective: Get ECN support in QUIC already from beginning
  - Implement necessary support for ECN
  - Congestion control based on ECN is separate work
  - Not a given that this draft becomes a WG item

- Feedback from : Marcelo Bagnulo Braun, Michael Welzl, Mirja Kühlewind, Niklas Widell, Koen De Schepper, Piers O'Hanlon, Brian Trammell, Bob Briscoe..

# OUTLINE OF
# DRAFT-JOHANSSON-QUIC-ECN

- QUIC specific:
  - ECN negotiation (or ECN capability sensing)
    - Performed after connection setup (at least for now)
  - ECN feedback : In ACK frames
  - Monitoring

- More general:
  - Fallback in case of ECN failure
  - OS sockets specifics

# ECN NEGOTIATION/CAPABILITY SENSING

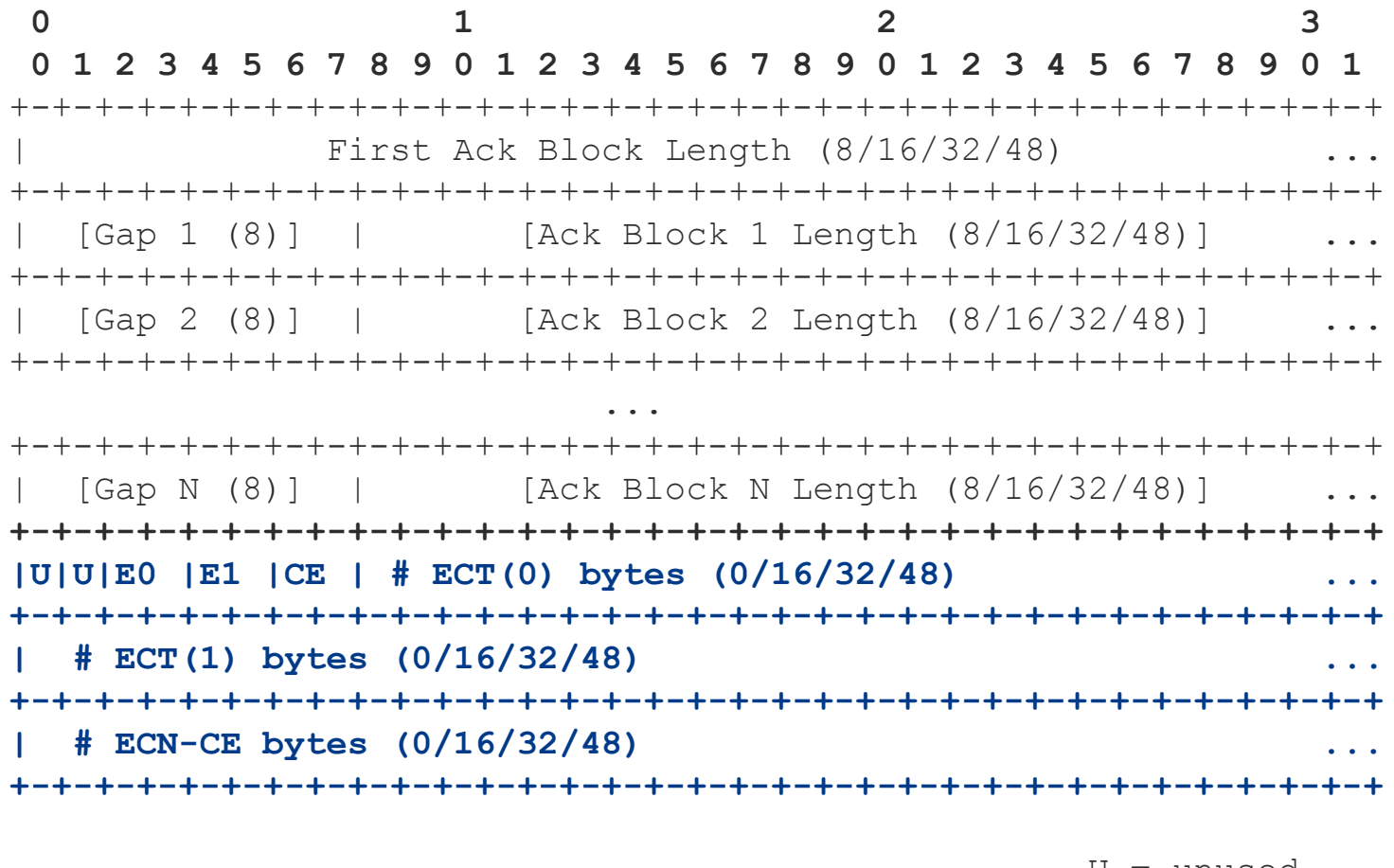
- Takes place after connection setup → avoid that ECN failures delay connection setup
  - No matter how unlikely this is…
  - Generalized ECN (bagnulo..) → ECN Negotiation already at connection setup possible ?
- Implemented as a 2 octet ECN negotiation frame
- Both peers send ECN negotiation frame  and echoes the ECN negotiation frame
  - ECN in one direction possible in some cases
- IP header ECN bits are set to ’11’ when ECN negotiation frames are transmitted
  - Or should it be ECT(0) or ECT(1) ?

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type           |C|R|W|U U U|E E|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- C: Challenge bit, indicates that the transmitted ECN negotiation frame is a challenge, if bit is not set then it is a response.
- R: Possible to read ECN bits in IP header
- W: Possible to write ECN bits in IP header
- EE : Echo of ECN bits
- U: Unused

# ECN ECHO

- Included in ACK frame
- E0, E1 and CE fields indicate encoding length of ECT(0), ECT(1) and CE marked bytes
  - 00: 0 bits
  - 01: 16bits
  - 10: 32bits
  - 11: 48bits
- Min overhead = 1 octet
- Possible to report ECN even though ECN is not negotiated
- How are bytes counted (recovery draft is silent on this) ?
  - QUIC
  - QUIC + UDP ?
  - QUIC + UDP + IP ?




U = unused

# OTHER

- Monitoring
  - Useful for indication of paths that do not implement ECN support correctly
  - Details T.B.D
- ECN fault detection and fallback
  - Details T.B.D but earlier work exists
- OS socket specifics
  - Document OS socket specifics i.e. access to ECN bits in IP header from user space

# WAY FORWARD

- Add ECN negotiation and ECN echo to draft-ietf-quic-transport
- Add ECN (classic) handling to  draft-ietf-quic-recovery

- L4S handling ?
  - Add specific details or placeholder ?

# COMMENTS WELCOME

- email : ingemar.s.johansson@ericsson.com