ECN for RTP over UDP/IP
draft-westerlund-avt-ecn-for-rtp-00.txt

Magnus Westerlund
Ingemar Johansson
Colin Perkins
Aims & Rationale

• Much real-time multimedia traffic uses RTP running over UDP/IP
  – Modern codecs are highly adaptive; good implementations of RTP react to packet loss as a congestion signal
  – But, impact of packet loss on real-time audio-visual flows is highly visible and disrupts user experience
  – ECN support would allow codec to adapt before loss occurs
    • The application controls how it reduces its sending rate, and hence how media quality is impacted
    • Better user experience than responding to packet loss, and more network friendly
ECN for RTP over UDP/IP

• Initially seems straight-forward:
  – Signal ECN support in SIP using SDP offer/answer
  – Set ECT on RTP data packets sent in UDP/IP
  – Send feedback piggybacked on RTCP reception reports
    • (No portable way to monitor received ECN marks on UDP)
  – Respond to ECN-CE by varying media encoding rate

• Yes, but…
Why is ECN for RTP Difficult? (1/5)

• Signalling
  – RTP relies on out-of-band signalling to initiate sessions; no in-band handshake or negotiation
    • i.e. no equivalent of TCP three-way handshake to negotiate ECN support
  – SIP can negotiate the end-point *capability* to support ECN, but says nothing about the media path
  – ICE can be extended to test the media path in *some* cases
Why is ECN for RTP Difficult? (2/5)

• Feedback
  – RTP does not explicitly acknowledge receipt of datagrams
  – RTCP provides reception quality feedback
    • Usual feedback interval $O(\text{seconds})$; but configurable
    • RTP/AVPF allows rapid feedback, provided feedback event rate within configured bandwidth constraint

  – Implies slower adaptation than TCP if congestion events are frequent
Why is ECN for RTP Difficult? (3/5)

• Congestion Response
  – Modern codecs can adapt over a wide rate, but often have constraints on what transmission rates are possible, and how quickly they can adapt
  – Frequent variation destroys user experience

  – Can respond to congestion, but unlikely to be TCP friendly (no worse than RTP over UDP/IP without ECN)
Why is ECN for RTP Difficult? (4/5)

- **Middle-boxes**
  - RTP explicitly supports application level *translators* and *mixers* within the network
    - Translator is a middle-box; must interpose itself in the ECN negotiation, split the connection, respond to congestion
    - Mixer acts as end-point; terminates transport connections
  - Only *part* of an RTP session may support ECN
Why is ECN for RTP Difficult? (5/5)

• Multicast
  – RTP is inherently a group communication protocol
    • ASM with many-to-many groups and multicast feedback
    • SSM with unicast feedback, potentially very large groups
      – IPTV channels, potentially millions of receivers

  – ECN per sender tree? For the entire group? All receivers?
    Again, only parts of the session may support ECN
  – May require receiver driven congestion response (layered coding?)
ECN for RTP over UDP/IP: Proposal

• Four pieces to the proposed solution:
  – Negotiation of ECN capability
    • SIP with SDP offer/answer; ICE option
  – Initiation and verification of ECT
    • Using RTP and RTCP
    • Using STUN and ICE
  – Ongoing use of ECN with RTP session
  – Failure detection, verification, and fallback
Initiation and Verification of ECT

• If end-points are capable, how to negotiate ECT?
  – Using RTP and RTCP
    • Mark a small fraction of RTP/UDP/IP packets as ECT during probing phase; don’t ECN mark any RTCP packets
    • New RTCP feedback packet reports receipt of ECT marked packets
    • Sender switches to using ECN for all RTP packets once the receiver population is stable, and all receivers report receipt of ECT marked packets
      – Choice to use ECN made on per-sender basis
      – Implications for multicast groups
  – Using STUN/ICE – see draft for details
Ongoing use of ECN with RTP

• RTCP reporting and feedback
  – Regular RTCP reports; use RTP/AVPF for CE events
  – ECN nonce + RLE or bit vector of lost/marked packets

• Congestion response
  – Sender driven, e.g. TFRC
  – Receiver driven, e.g. layered coding

• Detecting failure
  – Misbehaving receivers or middle-boxes
  – Path changes and/or mobility
  – Group membership changes

Continually monitor ECN operation and fallback to non-ECN mode if necessary
Input and Future Directions

• Technical details of RTP/RTCP extensions to be discussed in AVT on Friday
  – draft-westerlund-avt-ecn-for-rtp-00.txt

• From this group:
  – ECN for RTP over UDP/IP is a significant change compared to ECN for TCP/IP
  – Is this conceptually a good idea?
  – What transport issues have we missed?

  – Please read the draft!