ECN for RTP over UDP

draft-ietf-avt-ecn-for-rtp-03

Magnus Westerlund
Ingemar Johansson
Colin Perkins
Piers O’Hanlon
Ken Carlberg
Outline

› Introduction
› Changes
› Issues
  – Nonce
  – Initiation mechanisms
  – Combining ECN packets
  – Registry for ICE options
› Going Forward
Introduction

› Specifying a general mechanism for how to use ECN for RTP when sent over UDP
› Needs to work for all usages of RTP
  – Unicast
  – Multicast
  – Mixer and Translators
› Solution consists of phases
  – Signaling
  – Initiation
  – Ongoing usage
  – Fault detection and recovery
› Thanks to Bob Briscoe for extensive review of -02
Changes since previous version

› Removed Nonce (more about this later)
› Clarified some terminology
› Some restructuring moving RTCP Report Extensions prior to usage description
› Changed description of when Leap of Faith is applicable
› Suggestion that implementations should log ECN path failures found
› Assigned some code-points
  – RTP/AVPF transport layer feedback packet type 6
  – RTCP XR block type 13
› Various editorial cleanups
Issue – ECN Nonce

› Support for ECN nonce was removed in -03 following email discussion around anti-cheating mechanisms raised by Bob Briscoe

› Cumbersome mechanism with low utility
  – Big Report packets
  – Not fool proof

› Relies on sender being trustworthy and honest
  – This is likely true for infrastructure devices

› A Cheater can easily avoid the ECN nonce mechanism
  – Claim to be non-ECN compliant, and lie about packet loss

› Is there consensus on removing this from base solution?
  – Nonce mechanism could be revived later as an extension (although we need to be careful to ensure the signalling is extensible to allow this)
Issue – Initiation Mechanisms

Draft currently specifies 3 mechanisms for initiation:
- Probing using occasional ECN-marked RTP packets, with RTCP feedback
- Leap of Faith
- STUN-based probing for use with ICE

Cullen Jennings raised that he would prefer only one method.
Probing appears to be the only generally applicable mechanism (i.e., that works in all use cases):
- ICE not always used, and does not work in multicast
- Leap of faith can result in failure if a middle-box drops ECN-marked packets

Suggestion:
- MUST implement RTCP-based probing; leap-of-faith is OPTIONAL
- If ICE is implemented then MUST implement STUN-based probing, but MAY fall-back to RTCP-based probing if that fails
Issue – Combining ECN packets

› In Translators ECN packets can be split or joined – what happens to ECN-CE markings on such packets?

› Current solution based on TCP rules
   – If a marked packet is split, all resulting packets are marked; if a marked packet is combined, the resulting packet is marked
   – Bob Briscoe proposes marking based on the ratio of marked bytes
     › “…the outgoing packet SHOULD be ECN-CE marked with a probability proportional to the ratio of ECN-CE bytes to total bytes in the incoming packets being combined. However, for simplicity, the outgoing packet MAY be ECN-CE marked if any of the incoming packets are ECN-CE marked. This latter option will inflate the amount of congestion indicated, but it is at least safe.”

› Is the improved resolution in congestion volume worth the complexity?
Issue – ICE Options IANA Registry

› RFC 5245 states that an ICE options registry exists, but doesn’t define it
› This draft needs to register an option into this registry!

› Solutions:
  – Define the registry in this draft? (preferred)
  – Submit a separate draft to MMUSIC?
Going Forward

› Known issues still to address:
  – Add SDP parameter for XR and FB packets
  – Clarify SDP Offer/Answer and write SDP examples
  – Clarify what parts are mandatory to implement
  – Improve specification modularity
  – Fix SDP attribute so “ect” remains extensible
  – Improve considerations how media stream directionality and ECN directionality interact
  – Determine if optimization for end-points with multiple SSRCs in unicast can have quicker resolution of ECN capability

› Resolve the issues discussed today

› Aim at having next version of draft in December, hope to be ready for WG last call by IETF 80