ConEx Specification for Modification to TCP

ConEx – 79. IETF Beijing – 11th Nov. 2010

Mirja Kühlewind <mirja.kuehlewind@ikr.uni-stuttgart.de>

University of Stuttgart, Germany
Outline: ToC of the up-coming I-D

1. Introduction
   1.1. Overview of the ConEx Abstract Mechanisms
   1.2. Overview of the IPv6 Encoding

2. TCP Handshake Negotiation

3. Sender Modification
   3.1. Full ConEx Mode
   3.2. ECN-Co Mode
   3.3. Sender-only Mode
   3.4. Recommendation for ConEx Credits

4. Receiver (Optional Modifications)
TCP Handshake Negotiation (1)

Bytes 13 and 14 of the TCP Header:

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|               |           |
| N | C | E | U | A | P | R | S | F |
| Header Length | Reserved |
| S | W | C | R | C | S | S | Y | I |
|               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

**SYN**: Use TCP NS = 1 (ECN Nonce bit) to indicate ConEx support
- MUST/SHOULD request ECN support as well: NS = CWR = ECE = 1

**SYN/ACK**: Receiver is ConEx enable set CWR: CWR = 1, ECE = 0
- NS bit can be used to signal congestion information of SYN (if SYN ECN-enabled)
- CWR = ECE = 1 should not be used because buggy receiver might just echo SYN bits
- Otherwise receiver replies as specified for ECN: CWR = 0, ECE = 1 or CWR = ECE = 0
  → 3 ConEx modes: Receiver is **ConEx-enabled** or just **ECN-enabled** or **none** of both

- No changes to ECN (RFC 3168: The Adoption of ECN to IP)
- No support of ECN Nonce (RFC 3540: Robust ECN Signaling with Nonces)
TCP Handshake Negotiation (2)

Questions

- Should ConEx be bundled with ECN capability at the sender? Or is there a case for a request for drop-only ConEx?
- Should ConEx always set SACK-Permited Option in SYN (RFC 2018)?
- Use of NS in SYN/ACK? Enable ECN on SYN (with different coding) to avoid SYN packet drops?
- Middlebox issues?
Sender Modification

- Set ConEx bits in IP header (based on selected coding scheme in IPv6 draft)
- Monitor number of drops; number of ECN markings from receiver feedback (depending on mode)
- No modifications to TCP congestion control (required)

Questions

- Credit signal processing mandatory or just recommended?
- Handling of pure ACKS, retransmissions, window probes...?
- Should these related questions be discussed in this draft?
  - Requirements for dropper design based on credit processing + related security considerations
  - Congestion control/management
Changes only needed if receiver is ConEx enabled

- Monitor number of ECN markings

- Signal total number of ECN-markings to sender in TCP header of ACK
  - RFC 3168 (The Addition of ECN to IP): Receiver sets the ECE bit in every packet after observing a CE mark form the network until a data packet with the CWR bit set is received
    → Not more than one ECN congestion signal per RTT
  - Give feedback on all observed ECN-markings & be robust against loss of ACKs
  - Different mechanism for feedback encoding needed (by using the NS, CWR and ECE bit)

**Question**

- How encode ECN feedback in the ACK?
  - Max. number of two feedback markings per ACK (delayed ACKs -> RFC 2158: TCP Congestion Control)?
    → 3-bit counter (internal counter for the total number of seen CE markings, proposed by re-ECN to encode counter value modulo 8)
    → Signal additional number of ECN markings + ACK loss detection
    → Some kind of redundancy (sending the same information in subsequent ACKs)...
    → Other proposals???