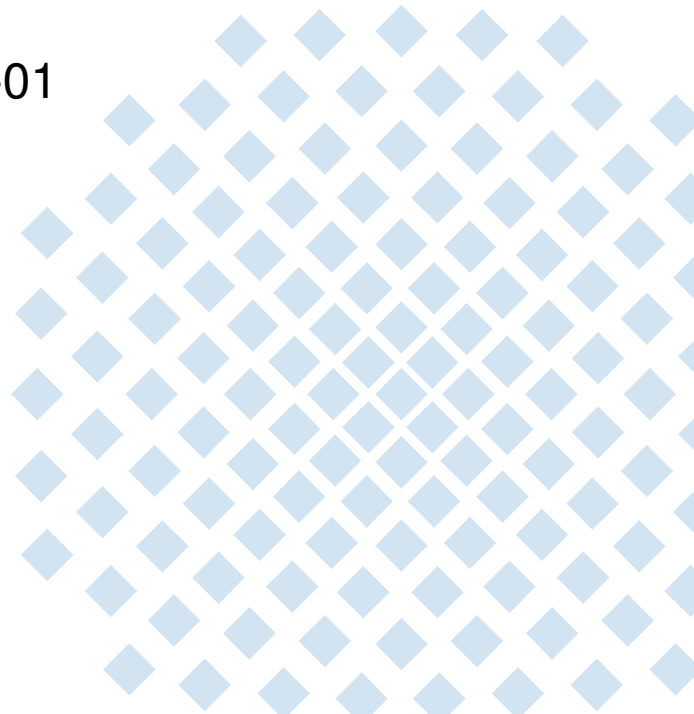
TCP modifications for Congestion Exposure

ConEx – 83. IETF Paris – March 29, 2012

draft-ietf-conex-tcp-modifications-01

Mirja Kühlewind <mirja.kuehlewind@ikr.uni-stuttgart.de>
Richard Scheffenegger <rs@netapp.com>



Congestion Accounting

- Estimation of loss information is available in bytes (# of ack'ed bytes and SACK)
- ECN refers to one or more packets marked per RTT
  - but ACK might ack more than one packet (# of ack'ed bytes only gives maximum of bytes)

→ Changed MUST to SHOULD

```
"A TCP sender SHOULD account congestion byte-wise..."
```

Timeliness of the ConEx Signals

- The sender SHOULD sent the ConEx signaling with the next available packet
- The sender MUST NOT delay the ConEx signal more than one RTT

Loss Accounting

- Loss exposure gauge (LEG) can get negative because of spurious loss detection
  "If the CEG or LEG counter is negative, the respective counter SHOULD be reset to zero within one RTT after it was decreased the last time or one RTT after recovery if no further congestion occurred."
- Slightly delay ConEx signal to handle spurious retransmission (if no SACK available)
  "If SACK is not available or SACK information has been reset for any reason, spurious retransmission are more likely. In this case it might be valuable to slightly delay the ConEx loss feedback until a spurious retransmission might be detected."

Other Changes

- Text added on how ECN works
- Refined calculation of bytes to be marked
- Added some text to Security Considerations sections
  - With advanced ECN compatibility modes no reliable feedback of congestion notification anymore

    → Influence on congestion control
  - But with persistent congestion the probability to receive a notification increases and in worst case loss occurs

    → This will avoid a congestion collapse

  → More text needed?
- Recommendations of credits in Slow Start (every 4th packet)
  but no further recommendations on credits
  - No further credits needed if standard TCP congestion control (reno) is used and congestion feedback from receiver is correct
  - Is a more general recommendation for other congestion control algorithms needed?

Open Issues

- Sender might send too few ConEx markings (if 'classic' ECN is used or ConEx markings get loss)
  - → audit might penalize the flow (-> drop additional packets)
  - → Is further action needed if losses are high?
  - → Should there be a recommendation in this draft?
  - → What is the right action to take?
  - → Nandita: What changes, if any, are required to accounting when ConEx marked packets get lost?
- Define DelieverdData (instead of acked_bytes) to correctly react to duplicated ACKs with and without SACK (if no new data has been ack'ed)

Backup

Accounting Congestion

ECN-based Congestion feedback

Congestion Exposure Gauge (CEG): num. of outstanding bytes with E bit

Accurate ECN feedback

→ **`CEG += min( SMSS*D, acked_bytes ); D = # of ECN feedback marks`**

Classic ECN support

1. Full compliance mode (Only one ECN feedback signal per RTT)

   → **`CEG += min(SMSS, acked_bytes)`** (whenever the ECE flag toggles from "0" to "1")

2. Simple compatibility mode
   - Set the CWR permanently to force the receiver to signal only one ECE per CE mark
   - Problem with delayed ACKs will cause information loss in high congestion situation
   - Proposed solution: Assume every received marking as M markings (M=2 delayed ACKs)

   → **`CEG += min(M*SMSS, acked_bytes + (M-1)*SMSS)`** (for every ECE flag)

3. Advanced compatibility mode
   - Set CWR only on those data segments, that will actually trigger an (delayed) ACK

   → if previous_marked: **`CEG += min( M*SSMS, acked_bytes )`**
   else: **`CEG += min(SMSS, acked_bytes)`**

Setting the ConEx IPv6 C(redit) Bit

From draft-ietf-conex-abstract-mech:

"The transport SHOULD signal sufficient credit in advance to cover any reasonably expected congestion during its feedback delay."

→ Credits should cover the increase of CWND per RTT

Slow Start (RFC5681 congestion control)

Exponential increase: double CWND very RTT

→ Number of credits has to be half of the flight size

→ Marking of every 4th packet (as credit which have been send in further RTTs are still valid)