More Accurate ECN Feedback in TCP

draft-ietf-tcpm-accurate-ecn-14

Bob Briscoe, Independent
Mirja Kühlewind, Ericsson
Richard Scheffenegger, NetApp

IETF-110 Mar 2021
Problem (Recap)
Congestion Existence, not Extent

- Explicit Congestion Notification (ECN)
  - routers/switches mark more packets as load grows
  - RFC3168 added ECN to IP and TCP

<table>
<thead>
<tr>
<th>IP-ECN</th>
<th>Codepoint</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>not-ECT</td>
<td>No ECN</td>
</tr>
<tr>
<td>10</td>
<td>ECT(0)</td>
<td>ECN-Capable Transport</td>
</tr>
<tr>
<td>01</td>
<td>ECT(1)</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>CE</td>
<td>Congestion Experienced</td>
</tr>
</tbody>
</table>

Port no’s, Seq no’s...

Data Offset | Res- | CE | E | U | A | P | R | S | Y | I | N | Window
            |     |    |   |   |   |   |   |   |   |   |   |              |
            |     |    |   |   |   |   |   |   |   |   |   |              |
Checksum    | Urgent Pointer

TCP Options...

- Problem with RFC3168 ECN feedback:
  - only one TCP feedback per RTT
  - rcvr repeats ECE flag for reliability, until sender's CWR flag acks it
  - suited TCP at the time – one congestion response per RTT
Solution (recap)
Congestion extent, not just existence

- AccECN: Change to TCP wire protocol
  - Repeated count of CE packets (ACE) - essential
  - and CE bytes (AccECN Option) – supplementary

- Key to congestion control for low queuing delay
  - 0.5 ms (vs. 5-15 ms) over public Internet

- Applicability: (see spare slide)
Field Order of AccECN TCP Option

- How to distinguish 2 different field orders in the AccECN Option
  - ExxB = Echo Byte counter xx, where xx = E0, E1, CE (each 3 B)

<table>
<thead>
<tr>
<th>kind</th>
<th>length</th>
<th>EE0B</th>
<th>[ECEB]</th>
<th>[EE1B]</th>
</tr>
</thead>
<tbody>
<tr>
<td>kind0</td>
<td></td>
<td>EE0B</td>
<td>[ECEB]</td>
<td>[EE1B]</td>
</tr>
<tr>
<td>kind1</td>
<td></td>
<td>EE1B</td>
<td>[ECEB]</td>
<td>[EE0B]</td>
</tr>
</tbody>
</table>

- After IETF-109, a third alternative:
  1) Two Option Kinds [MScharf]
  2) Add flags byte to option [Ilpo]
  3) Use most significant bit of first 24-bit field to signal field order [Joe]

- Conclusion
  - Kept two Option Kinds after a little push-back against #3
Forward Compatibility vs. Covert Channel

<table>
<thead>
<tr>
<th>kind</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3 octets]</td>
<td>[3 octets]</td>
</tr>
<tr>
<td>[3 octets]</td>
<td></td>
</tr>
</tbody>
</table>

- **Background**: Valid AccECN Option lengths: 2 + (0, 3, 6, or 9) octets
  - For forward compatibility, if the AccECN Option is of any other length, implementations MUST use those whole 3-octet fields that fit within the length and ignore the remainder of the option, treating it as padding.
  - A middlebox **claiming to be transparent** at the transport layer MUST forward the AccECN TCP Option unaltered, whether or not the length value matches one of those specified.

- Creates a covert channel of up to 29B [MScharf]
  - Now identified in Security Considerations
  - Prompted chairs to ask for early SECDIR review

- We could sacrifice forward compatibility; but no real need here

- Not a new covert channel
  - A TCP MUST ignore without error any TCP option it does not implement [RFC1122]

- Where nec., current IDSs already close off these channels
  - block unknown options or known options with unknown lengths
To ACK ACKs or not to ACK?
That is the question

An AccECN Data Receiver:

- SHOULD immediately send an ACK whenever a data packet marked CE arrives after the previous packet was not CE.
- MUST immediately send an ACK once 'n' CE marks have arrived since the previous ACK, where 'n' SHOULD be 2 and MUST be in the range 2 to 6 inclusive.

- Intentions:
  - rapid feedback at congestion onset
  - reduce risk of double wrap of 3-bit ACE counter

- Realized 2nd bullet could lead to ACKs of ACKs (first bullet deliberately doesn't)
  - 'OK in principle': ACKing new information (new CE marks)
  - to maintain cwnd during idles, or ready for adding ACK CC
  - but potential ACK ping-pong must be strongly damped
To ACK ACKs or not to ACK?

DupACKs is another question

- ACKs of ACKs could look like DupACKs [Yoshi]
  - If ACK stream CE marked
  - and data volleys take turns

- Low risk
  - Already a corner case
  - and only if SACK not negotiated*
  - harm would be spurious re-xmt(s)

* AccECN recommends SACK. If SACK-negotiated, and if no SACK on ACK, not a DupACK
To ACK ACKs or not to ACK?
What is the answer?

Two positions:

A) Prevent ACKs of ACKS completely

MUST immediately send an ACK once 'n' CE marks have
arrived since the previous ACK and there is
outstanding data to acknowledge, where 'n' SHOULD be 2
and MUST be in the range 2 to 6 inclusive.

B) Take opportunity to still feed back CE on ACKs,
but damp any potential ACK ping-pong

• MUST immediately send an ACK once 'n' CE marks have
  arrived since the previous ACK {1}, where 'n' SHOULD
  be $\frac{2}{3}$ and MUST be in the range $\frac{2}{3}$ to 6 inclusive.

• There are simplicity arguments on both sides
Other changes

- Editorial fixes throughout
  - Esp. ACK Filtering
  - thx to Gorry's latest review
Status & Next Steps
draft-ietf-tcpm-accurate-ecn-14

- Once ACKs of ACKs resolved ready for WGLC
- draft-ietf-tcpm-generalized-ecn (EXP) dependent on this
- April’20 tcpm interim:
  - WG resolved to wait a while for L4S, but go ahead soon if still waiting
AccECN

Q&A

spare slides