Speeding Up TCP with Selective Loss Prevention

Zhenyu Zhou*, Xiaowei Yang

Duke University
1s = ?
Page Load Time Matters

1s = $1B

The “Narrow Waist”

https://www.systemsapproach.org/blog-archive/http-is-the-new-narrow-waist
The “Narrow Waist”

https://www.systemsapproach.org/blog-archive/http-is-the-new-narrow-waist
Improving TCP Latency

- DCTCP [SIGCOMM’10]
- SPDY [Chromium Projects]
- QUIC [SIGCOMM’17]
- NDP [SIGCOMM’17]
...

Improving TCP Latency

• Previous work on congestion control
  • Fully utilize available bandwidth

• An orthogonal view
  • What is the main contributor of TCP latency?
Improving TCP Latency

• Previous work on congestion control
  • Fully utilize available bandwidth

• An orthogonal view
  • What is the main contributor of TCP latency?

Timeout
TCP Timeout

• Classic alleviation: Fast retransmission
  • Not always triggered
  • Our observation: timeouts account for ~10.1% of retransmission events

• Takeaway
  • TCP timeout still significantly contributes to TCP latency
Goal

Improving TCP latency by reducing TCP timeout events
Selective Loss Prevention

- Our solution: Selective Loss Prevention (SLP)
  - Predicate “important” packets more likely to cause timeouts
  - Aggressively duplicate these “important” packets

- Why selective duplication?
  - **Automatic Repeat Request vs. Forward Error Correction**
  - TCP applies ARQ (retransmission)
  - Duplication is the simplest form of FEC
  - Applying FEC to all packets is ineffective (analysis in paper)
Challenges

• How to determine the “important” packets?

• How to prevent redundant packets from congesting the network?
Challenges

• How to determine the “important” packets?
  • Analyze packet traces from DCN and Internet

• How to prevent redundant packets from congesting the network?
  • Co-design congestion control algorithm with duplicate packets
Important Packets

• Position analysis
  • Fraction of timeout-based retransmission for packets lost at different *positions* in a TCP connection

(a) Forward half from client to server (b) Reverse half from client to server

(c) Foward half from server to client (d) Reverse half from server to client
Important Packets

- Position analysis
  - Fraction of timeout-based retransmission for packets lost at different *positions* in a TCP connection

- Takeaway
  - Head and tail losses are more likely to cause timeouts
  - Especially, more than 70% of TCP SYN packet losses caused a timeout
Important Packets

- Flag Analysis
  - Fraction of timeout-based retransmission for packets lost with different TCP flags
Important Packets

• Flag Analysis
  • Fraction of timeout-based retransmission for packets lost with different TCP flags

• Takeaway
  • Packets near the beginning or end of a connection, and packets carrying PSH flag are more likely to cause timeouts
Incorporate SLP into TCP

tcp_send(pkt)

SLP?

YES

Packet Duplicator

NO

Sliding Win Allowed?

YES

Pacing

NO

Normal TCP Processing
Incorporate SLP into TCP

- Only duplicate packets when beneficial
- Adjust sliding window to avoid further congestion

Formal analysis can be found in the paper
Preliminary Evaluation

File completion time

(a) Without SLP

(b) With SLP
Thanks for your attention!

Questions?