A Proposal for SLUMS

Venkata N. Padmanabhan
Microsoft Research

www.research.microsoft.com/~padmanab

SLUMS BOF, 44th IETF
March 1999
Overview

• TCP can satisfy many of SLUMS goals
  – make TCP connection cheap enough that applications can use as many as they would like
  – ALF at the granularity of TCP connections
    • each connec provides a logically-independent byte stream

• Benefits
  – minimal change to existing protocol and API
  – relieves applications from being constantly engaged in transmission/retransmission of data
Problems

- Cost of connection setup
- Large packet count
- State storage and management overhead
- Short connections perform poorly
- Concurrent connections compete
- Strict ordering of single connection is restrictive
- Cost of connection setup
Transaction TCP

- TCP accelerated open eliminates RTT for setup
- But opens up security holes
- Expand CC cache to include key info
  - security & performance at the expense of extra state
  - trade-off exists even with UDP

- T/TCP also helps cut down packet count
  - 3 packets for minimal transaction
Challenges of Short and/or Concurrent Connections

• Concurrent connections compete
  – independent probing ⇒ repeated slow start
  – increased packet loss rate
  – arbitrary bandwidth sharing beyond applic control

• Dominance of timeouts [BPS+98]
  – insufficient dupacks to trigger fast retransmission

• Slow start penalty
  – RFC-2140, RBP [VH97], TCP fast start [Pad98]
  – out of scope of SLUMS
TCP Session [Pad98, BPS+98]

• Decouple 2 components of TCP functionality
  – reliable, ordered byte-stream service: per connection
  – congestion ctrl/loss recovery algorithms: per session

• Three components
  – integrated congestion control
  – connection scheduling
  – integrated loss recovery
Integrated Congestion Control and Connection Scheduling

- Single congestion window for entire session
  - sender entitled to send when $ownd < cwnd$
  - sender can choose to send on any connection
  - independent flow control

- Connection scheduling
  - hierarchical round-robin (HRR) [KKK90]
  - $setwt()$ and $resetwt()$ to dynamically vary weights
  - other schedulers can certainly be used
  - can potentially interface with RSVP/diffserv
Integrated Loss Recovery

- Pool together pkt delivery info across conns to make data-driven loss recovery more effective
  - use *later* acks in addition to dupacks
  - need to be careful with delayed acks
- Loss recovery rules for a connection
  - at least 1 dupack + 3 dup/later acks for a segment
  - at least 3 dup/later acks for at least 2 segments
- Rtx timeout only if all acks streams have stalled
- 7-10X reduction in # rtx timeouts [Pad98]
Ack Aggregation

• Ack loss \(\Rightarrow\) false retransmission possible
  – but experiments in [Pad98] do not exhibit this problem

• To be safe, aggregate acks
  – TCP option to carry ack info for other connections
  – 2 bytes of kind/length + 8 bytes of port/ack number
  – up to 4 such “acks” per packet
  – either in place of or in addition to regular acks
  – helps reduce packet count
Efficient State Management

- **TCP Session**
  - cong ctrl/loss recovery variables in SCB
    - 28 bytes out of 134 bytes in TCB move to SCB
  - only one retransmit timer per session
- **Much smaller TCB for inactive connections**
- **Better demultiplexing algorithms [Mog95]**
  - use hashing instead of linear search
  - maintain TCBs of active connections separately from those for inactive connections
Summary

- Cheap connections ⇒ applic could implement ALF at the granularity of connections
- Connection scheduling to reflect priorities
- Optimized TCP with minimal protocol/API mods helps address many of SLUMS goals
  - quick setup, ALF, independent flow control, multiplexing, QoS consciousness between the streams, integrated congestion control, avoiding repeated slow start, ack aggregation, reduced state management overhead
Limitations

- No failover upon change in IP address
  - Mobile IP style tunneling is a possibility but would be inefficient
  - IP option to carry unique host ID?

- TCP provides enforces reliability
  - selective reliability possible at the granularity of connections