An Overview of New TCP-ish Protocols

Matt Zekauskas, matt@internet2.edu

JET Roadmap Workshop

13 April 2004

Caveat

- I follow work on TCP, but I probably don't have exhaustive knowledge (esp. with Mathis, Cottrell in audience!)
- This is meant to give an overview, and stimulate discussion
- So, if I'm missing something, speak up!

Some References

- Start with PFLDnet 2004 Workshop material

  http://www-didc.lbl.gov/PFLDnet2004/program.htm

- Bullot/Cottrell/Hughes-Jones (S5):
  - Evaluating a bunch of stacks
- FAST
- XCP

People are Tweaking Reno TCP for LFNs

- Ramps up too slowly
  - Change response function [slow-start, parameters]
  - Increase MTU
- Requires infinitesimal loss (ramp after loss)
  - Change response function [AIMD, parameters]
  - Increase MTU
  - Parallel stream
  - Avoid loss via ECN
  - Pacing
  - Introduce delay sensitivity
  - Intimate knowledge of congestion from routers

Change Response Function

- Net100 WAD
  - tweak many parameters, also replace function
- Limited Slow-Start (RFC 3742)
  - Be gentler when # segments in flight large
- HS-TCP (RFC 3649)
  - Change AIMD parameters when # segments in flight large (plays well under congested conditions)
- S-TCP Scalable, Tom Kelley
  - Change parameters always, more agressive

Change Response Function

- Bic-TCP (binary increase cong. ctl.)
  - "correct RTT unfairness" – algorithmically adjust
- HS-TCP-LP
  - Combine HS-TCP with TCP-LP, and a less-than-best-effort service like Scavenger service
- Parallel TCP
  - Multiple streams
    conventional wisdom switching to "too aggressive"
- There are others… H-TCP, TCP Westwood, …

Adding Delay Sensitivity

- Old: TCP-Vegas
  - Conventional wisdom: doesn't play well
- New: FAST
  - Base on control theory, goal is fairness at flow level rather than packet level behavior
  - Maintain small queues
  - Quickly get to equilibrium point, then move slowly
  - ? Respond to ack as well as forward
  - ? What if queue full, that's also constant delay
  - ? Playing with Reno TCP? [So far: good, as long as FAST is filling bottleneck…]

XCP in a Nutshell

- XCP is a new congestion control protocol developed by Dina Katabi
- End-systems tell routers what throughput they'd like to send at
- Routers make a per-flow allocation, inspect incoming packets, and reduce the throughput request to match the allocation (if necessary)
- [Slide blatantly stolen from Aaron Falk's PFLDnet presentation]

XCP promise

- Fills bottleneck rapidly
- Converges to fair allocation
- Good bottleneck link utilization
- Maintain small queues
  - [Parapharased from Aaron Falk's presentation at PFLDnet 2004]
- …but complete upgrade

Implications

- XCP too far out, but perhaps host experiments by others
- FAST is promising, but work continues
- We are a good testbed
- Encourage (controlled) testing?


INTERNET®
www.internet2.edu