TCP Friendly is an Oxymoron

Matt Mathis
rmcat BOF
IETF 84
2 Aug 2012

High level view

- TCP requires queues for correct operation
- RT traffic cannot tolerate queues
- The only solution is traffic segregation
  - Implies QoS or something

It is known that RT CC can't fix:

- No AQM - Buffer bloat
  - Bad all around
- Weak or misstuned AQM + high load
  - Bad all around
- Ideal AQM + medium RTT TCP slowstart
  - Good queue required to generate TCP's clock
- Ideal AQM + long RTT AIMD sawtooth
  - Good queue required to absorb peaks
- Ledbat with 100 mS threshold
  - Oops, could have been better
- Streaming video

Streaming video

- Application pacing
  - Read or write "copy buffer" on a timer
  - Every 5-10 seconds typical
  - 65kB-4MB depending on video quality
- With enough bandwidth or queue space:
  - lossless
  - cwnd rises above burst size
  - bursts go out at server interface rate
- We can fix this...

Trickle: Rate Limiting YouTube Video Streaming

Monia Ghobadi, Yuchung Cheng, Ankur Jain, Matt Mathis

USENIX ATC, June 2012

- Adaptive cwnd clamp to spread bursts
- No effect on user experience
- Improve ephemerial statistics
  - RTT statistics
  - Loss rate, but only in corner cases
- Risks to deployment
  - e.g. Impair adaptive quality selection, etc

Observations

- TCP creates queues
  - Always explicitly permitted
  - Sometimes even an explicit goal
- RT traffic sharing a queue with TCP
  - Is **always** at risk
  - It only works if TCP is otherwise throttled
- RT only works if the queue is mostly empty
  - This is an easily created corner case
    - Keep other user off of the network
    - constrain applications
    - But these are just workarounds, not real soln's
- RT really requires traffic segregation
  - e.g. QoS with separate queues

What happened to QoS?

- A decade ago all TCPs were lame
  - Cheaper to outbuild the load than deploy QoS
- Modern stacks are not lame
  - **Every** connection between **any** pair of hosts always does at least one of:
    - Raise the RTT
    - Raise the loss rate
    - Run out of data
  - **It is not possible to out build the load**
- So now we need QoS yesterday
  - This conversation is (mostly) out of scope here
  - But we can help

Expose the problem

- Build standard metrics into codecs
  - e.g. See RFC 5481
  - Must also distill into a simple quality parameter
- ISPs can benchmark with same metric
- Users can compare values
  - Bring market pressure to ISPs

RT CC design

- Assume (mostly) empty queue
- Estimate ("fair") RT capacity
  - Avoid self inflicted queueing delay
- If there is too much cross traffic
  - TFRC or a circuit breaker are all you can do
  - Note that you cannot avoid adding to the queue
- Don't over optimize the failure cases
  - e.g. a "better too much" estimator
  - e.g. a "better TFRC"
  - We want them to go away, not work around

Closing

Protecting TCP is unnecessary, it will not return the favor.

- Andrew McGregor