Implications on Upper Layers

draft-thaler-ip-model-evolution-01.txt

Dave Thaler

Context

- For purposes of this presentation “apps” is shorthand for anything above network layer
- Many “Apps” have embedded assumptions (or myths, increasingly…)
- Making them less true can break apps
- Making them more true can “fix” apps
- Let’s look at a few that are relevant to LISP and friends
  - See draft or INTAREA meeting for more

E2E delay of first packet to a destination is typical

- Examples of behavior:
  - Applications “ping” candidate servers and use the first one to respond
- Status:
  - PIM-SM, MSDP, MIPv6, etc allow deterministic path switching during initial data burst
  - “Choice” of server can hence be highly non-optimal, resulting in longer paths, lower throughput, and higher load on the Internet

Reordering is rare

- Examples of behavior:
  - Some firewalls/NATs assume initial fragment arrives first, results in packet loss
  - TCP enters fast retransmit if 3 packets arrive before a late packet
  - Reordering increases buffering requirements (and jitter) in many apps
- Status:
  - Per packet load balancing in some places
  - Some hosts send last fragment first
  - Deterministic path switching protocols cause reordering among initial packets

Loss is rare and probabilistic, not deterministic

- Examples of behavior:
  - Applications “ping” candidate servers and use the first one to respond
  - Bursty source applications (including ones that result in fragmentation)
- Status:
  - “Wake-on-LAN” cards drop initial packet(s)
  - Some firewalls drop due to fragment reordering
  - Some RRG, MANET, etc proposals result in queuing initial packets, resulting in loss as queue overflows
  - This happens with ARP/ND too, but only over 1 hop so generally not observable
  - MSDP says forwarding initial packets are optional
  - Cascading *multiple* of the above makes it even worse

An "address" used by an application is the same as the "address" used for routing

- A.k.a. “ID == Locator”

- Examples of behavior:
  - Apps make assumptions about locality (e.g., same subnet) by comparing addresses
  - Server-selection apps/protocols make assumptions about locality by comparing source address against configured ranges
  - Apps use raw sockets to read/write packet headers

- Status:
  - Not true with tunneling, most ID-locator split schemes, etc.
    - ID-locator split schemes like LISP only break it in the core of the Internet so only affects apps running there