15-744: Computer Networking

L-24 Network Measurements

Network Measurements

- How is the Internet holding up?
- Assigned reading
  - [Pax97] End-to-End Internet Packet Dynamics

Motivation

- Answers many questions
  - How does the Internet really operate?
  - Is it working efficiently?
  - How will trends affect its operation?
  - How should future protocols be designed?
- Aren't simulation and analysis enough?
  - We really don't know what to simulate or analyze
    - Need to understand how Internet is being used!
  - Too difficult to analyze or simulate parts we do understand

Measurement Methodologies

- Active tests – probe the network and see how it responds
  - Must be careful to ensure that your probes only measure desired information (and without bias)
  - Labovitz routing behavior – add and withdraw routes and see how BGP behaves
  - Paxson packet dynamics – perform transfers and record behavior
  - Bolot delay & loss – record behavior of UDP probes
- Passive tests – measure existing behavior
  - Must be careful not to perturb network
  - Labovitz BGP anamolies – record all BGP exchanges
  - Paxson routing behavior – perform traceroute between hosts
  - Lelan self-similarity – record ethernet traffic

Traces Characteristics

- Some available at http://ita.ee.lbl.gov
  - E.g. tcpdump files and HTTP logs
  - Public ones tend to be old (2+ years)
  - Privacy concerns tend to reduce useful content
- Paxson's test data
  - Network Probe Daemon (NPD) – performs transfers & traceroutes, records packet traces
  - Approximately 20-40 sites participated in various NPD based studies
  - The number of "paths" tested by NPD framework scaled with (number of hosts)^2
    - 20-40 hosts = 400-1600 paths!

Observations – Routing Pathologies

- Observations from traceroute between NPDs
- Routing loops
  - Types – forwarding loops, control information loop (count-to-infinity) and traceroute loop (can be either forwarding loop or route change)
  - Routing protocols should prevent loops from persisting
  - Fall into short-term (< 3hrs) and long-term (> 12 hrs) duration
  - Some loops spanned multiple BGP hops! → seem to be a result of static routes
- Erroneous routing – Rare but saw a US-UK route that went through Isreal → can't really trust where packets may go!

Observations – Routing Pathologies

- Route change between traceroutes
  - Associated outages have bimodal duration distribution
    - Perhaps due to the difference in addition/removal of link in routing protocols
- Temporary outages
  - Traceroute probes (1-2%) experienced > 30sec outages
  - Outage likelihood strongly correlated with time of day/load
- Most pathologies seem to be getting worse over time

Observations – Routing Stability

- Prevalence – how likely are you to encounter a given route
  - In general, paths have a single primary route
  - For 50% of paths, single route was present 82% of the time
- Persistence – how long does a given route last
  - Hard to measure – what if route changes and changes back between samples?
  - Look at 3 different time scales
    - Seconds/minutes→ load-balancing flutter & tightly coupled routers
    - 10's of Minutes → infrequently observed
    - Hours → 2/3 of all routes, long lived routes typically lasted several days

Observations – Re-ordering

- 12-36% of transfers had re-ordering
- 1-2% of packets were re-ordered
- Very much dependent on path
  - Some sites had large amount of re-ordering
  - Forward and reverse path may have different amounts
- Impact → ordering used to detect loss
  - TCP uses re-order of 3 packets as heuristic
  - Decrease in threshold would cause many "bad" rexmits
    - But would increase rexmit opportunities by 65-70%
  - A combination of delay and lower threshold would be satisfactory though → maybe Vegas would work well!

Observations – Packet Oddities

- Replication
  - Internet does not provide "at most once" delivery
  - Replication occurs rarely
  - Possible causes → link-layer rexmits, misconfigured bridges
- Corruption
  - Checksums on packets are typically weak
    - 16-bit in TCP/UDP → miss 1/64K errors
  - Approx. 1/5000 packets get corrupted
  - 1/3million packets are probably accepted with errors!

Observations – Bottleneck Bandwidth

- Typical technique, packet pair, has several weaknesses
  - Out-of-order delivery → pair likely used different paths
  - Clock resolution → 10msec clock and 512 byte packets limit estimate to 51.2 KBps
  - Changes in BW
  - Multi-channel links → packets are not queued behind each other
- Solution – Packet Bunch Mode (PBM)
  - Send a group of packets and analyze modes of different bunch sizes

Observations – Loss Rates

- Ack losses vs. data losses
  - TCP adapts data transmission to avoid loss
  - No similar effect for acks → Ack losses reflect Internet loss rates more accurately (however, not a major factor in measurements)
- 52% of transfers had no loss (quiescent periods)
- 2.7% loss rate in 12/94 and 5.2% in 11/95
  - Loss rate for "busy" periods = 5.6 & 8.7%
- Losses tend to be very bursty
  - Unconditional loss prob = 2 - 3%
  - Conditional loss prob = 20 - 50%
  - Duration of "outages" vary across many orders of magnitude (pareto distributed)

Observations – TCP Behavior

- Recorded every packet sent to Web server for 1996 Olympics
  - Can re-create outgoing data based on TCP behavior → must use some heuristics to identify timeouts, etc.
- How is TCP used clients and how does TCP recover from losses
  - Lots of small transfers done in parallel

Observations – TCP Behavior

Trace Statistic	Value	%Age
Total connections	1,650,103	100
With packet reordering	97,036	6
With rcvr window bottleneck	233,906	14
Total packets	7,821,638	100
During slow start	6,662,050	85
Slow start packets lost	354,566	6
During congestion avoidance	1,159,588	15
Congestion avoidance loss	82,181	7
Total retransmissions	857,142	100
Fast retransmissions	375,306	44
Slow start following timeout	59,811	7
Coarse timeouts	422,025	49
Avoidable with SACK	18,713	4

Observations – Self-Similarity

- Let X be a sequence of values drawn from a distribution
  - X is covariance stationary or wide-sense stationary (WSS) iff:
    - Mean does not change with time
    - Variance does on change with time
    - Autocorrelation is only a function of ?T
- WSS != stationary
  - Stationary requires that all X are drawn from same distribution
- Basic assumption of paper is that Ethernet bandwidth is WSS

Observations – Self-Similarity

- A self-similar process looks similar across many different time scales
  - Above hours, human behavior has significant effect
  - Poisson processes tend to smooth out
- Suppose that original X's were replaced by blocked version
  - Replace m consecutive samples of X with a single average value → $X^{(m)}$
- X is self-similar if:
  - Variance($X^{(m)}$) is slowly decaying as a function of m
  - Autocorrelation of $X^{(m)}$ is the same as X

Observations – Self-Similarity

- Variance($X^{(m)}$) is slowly decaying as a function of m
  - Implication → process has a heavy tail since tail probabilities do not fall (I.e. large variance)
- Autocorrelation decays slowly
  - Autocorrelation goes with k^{-B} (i.e. hyperbolically)
  - Termed long-range dependence

Observations – Self-Similarity Tests

- Variance-time plots
  - For each block size *m* calculate variance
  - Plot variance vs. *m* on log-log scale
  - If process is self-similar, fit line and slope will be related to Hurst parameter → –2 x (1 - H)
- R/S statistic
  - Calculate S^2, sample variance of $X_1 \ldots X_n$
  - R = Range = max(0, $W_1, W_2, \ldots W_n$) - min(0, $W_1, W_2, \ldots W_n$) where $W_k = X_1 + X_2 \ldots + X_k - kX_{avg}$
  - R/S should be proportional to n^H then it is self-similar

Other Motivations

- Can also measure current state of network to provide status and short-term predictions
- Need on-line real-time analysis of traffic and conditions
- Example systems include IDMAP, Remos, Sonar, SPAND

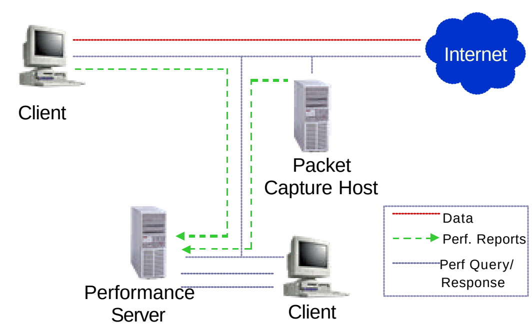
SPAND Assumptions

- *Geographic Stability:* Performance observed by nearby clients is similar → works within a domain
- *Amount of Sharing:* Multiple clients within domain access same destinations within reasonable time period → strong locality exists
- *Temporal Stability:* Recent measurements are indicative of future performance → true for 10's of minutes

SPAND Design Choices

- Measurements are *shared*
  - Hosts share performance information by placing it in a per-domain repository
- Measurements are *passive*
  - Application-to-application traffic is used to measure network performance
- Measurements are *application-specific*
  - When possible, measure application response time, not bandwidth, latency, hop count, etc.

SPAND Architecture



Measurement Summary

- Internet is a large and heterogeneous
  - There is no "typical" behavior → each path or region may be very different
  - Protocols must be able to handle this
- Internet changes quickly
  - New applications change the way the network is used
  - Some invariants remain across these changes

Beginning of Semester Objectives

- Understand the state-of-the-art in network protocols, architectures and applications
- Understand how networking research is done
- Training network programmers vs. training network researchers

Overview (1)

- Fast forwarding/routing
  - Typical structure of a router → where are the bottlenecks
  - Challenge of doing fast route lookup/packet classification → reduce memory lookups
- Routing protocols
  - Structure of the Internet
  - Routing protocols that match administrative structure
- Overlay routing
  - New approach to adding functionality to Internet
  - Key challenge of routing at a layer above
- Mobile routing
  - Routing without addressing structure (Mobile IP and ad-hoc)

Overview (2)

- Transport reliability
  - Techniques for loss recovery and tradeoffs between techniques
- Congestion control
  - Why is AIMD the right choice
  - How does TCP perform cong. ctl. and resulting performance
- Transport alternatives
  - Why is AIMD not always the right choice ☺
- Mobile transport
  - Why are wireless links are hard on transport
- Active queue managment
  - State-of-art in no per-flow state AQM → RED & Blue
  - Fair-queuing – how to implement and what it's good for

Overview (3)

- DNS
  - How it works and how it is used today
- Multicast
  - Techniques used to make multicast IP routing possible
  - Challenges that multicast create for upper layer protocols
    - Reliability, congestion control, address allocation, etc.
- QoS
  - How to provide guaranteed performance (Intserv) to individual flows and associated problems with scalability
  - How to signal performance requirements to network
  - How to provide more scalable (aggregated) service differentiation (DiffServ)

Overview (4)

- Different forms of network applications
  - HTTP – how usage patterns can impact design
  - CDNs – how to create scalable managed services
  - P2P – how to create scalable unmanaged services
- Security
  - Weaknesses in IP architecture and how to protect them
  - Why we need security infrastructure (firewall, certificate authorities, etc.)
- Measurement
  - Why we need to do this
  - What can we discover
- Design philosophy
  - Good to revisit some of the philosophy papers and examine how they impacted design

THE END!

- Networking has a wide variety of interesting topic areas
- Hopefully you should be able to pick up any networking research paper and understand both their motivation and methodology