

Avoiding Network Capacity Collapse

John Kristoff

jtk@depaul.edu

+1 312 362- 5878

DePaul University

Chicago, IL 60604

Capacity Collapse

- Scarcity of capacity
- (Dropped Traffic / Offered Traffic) increases
- *Goodput* decreases (approaches zero)
- Response time increases
- Very little or no real work gets done

Statistical Multiplexing

- A primary advantage of data networks
- Available capacity can be used by anyone
- Share capacity on first in, first out basis
- Build network based on average usage
- But IP is arbitrarily bursty
- Hence, will probably have some congestion
- How do you prevent capacity collapse?

IP Type of Service Field

"The Type of Service provides an indication of the abstract parameters of the quality of service desired." - RFC 791, September 1981

Twenty years later and still no Internet QoS!

TCP Congestion Avoidance

- TCP cannot *control* congestion
- It can *react* based on implicit network signals
- Assumes packet loss is due to congestion
- TCP is quite good - maybe too good
 - Tries to fully utilize network - it can go very fast
- Want TCP to go slow? Drop packets!
- Dropping packets reduces *goodput*

ICMP, UDP and Multicast

- Some protocols unresponsive to congestion
- Luckily TCP accounts for ~90% of the traffic
- Congestion *control* is needed
 - A function of the network
- How do we do it? is the question
 - RED and ECN
 - Scheduling and rate limiting
 - Price incentives

What about IPv6, ATM, MPLS...

- Are these ubiquitous in your network?
- Thought so.
- Probably wouldn't be a panacea anyway
- Next slide please...

(D)DoS Attacks are Related

- But we won't talk specifically about them
- Congestion control ideas may help us
 - With some added features
- Probably only a temporary capacity collapse
- Include this in capacity management plan

Let's Get More Capacity

- LAN capacity is cheap, we can overprovision
- Leased WAN links can be costly
- Internet service can definitely be costly
- Operational versus capital costs
- Ugh... provisioning problems and lead times
- Need simple, cheap and fast
- We only get to pick one, maybe two if lucky

Access Blocking

- DNS black holing
- IP router filters
- Null routes
- Site blocking

Rate Limiting

- IP, UDP and TCP based - usually
- IP addresses
- Protocol ports
- Strict limits
- Dynamic limits

UIUC Rate Limiting Experiment

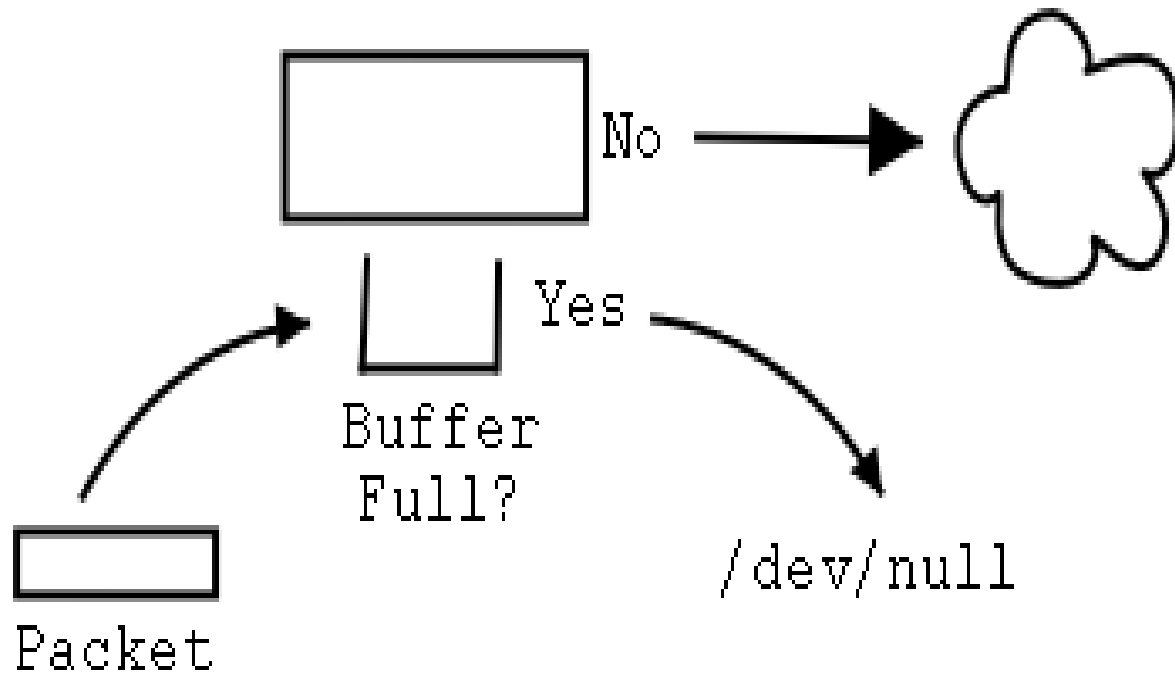
- Allow full capacity access by default
- "Out- of- profile" users are rate limited
- Increasingly aggressive limits if necessary
- Analyzing cflowd data to determine usage
- Dynamically upload CAR configs once/hour
- Scaling issues - a tad scary

http://www.ncne.nlanr.net/training/techs/2001/0128/presentations/2000101- kline1_files/v3_document.htm

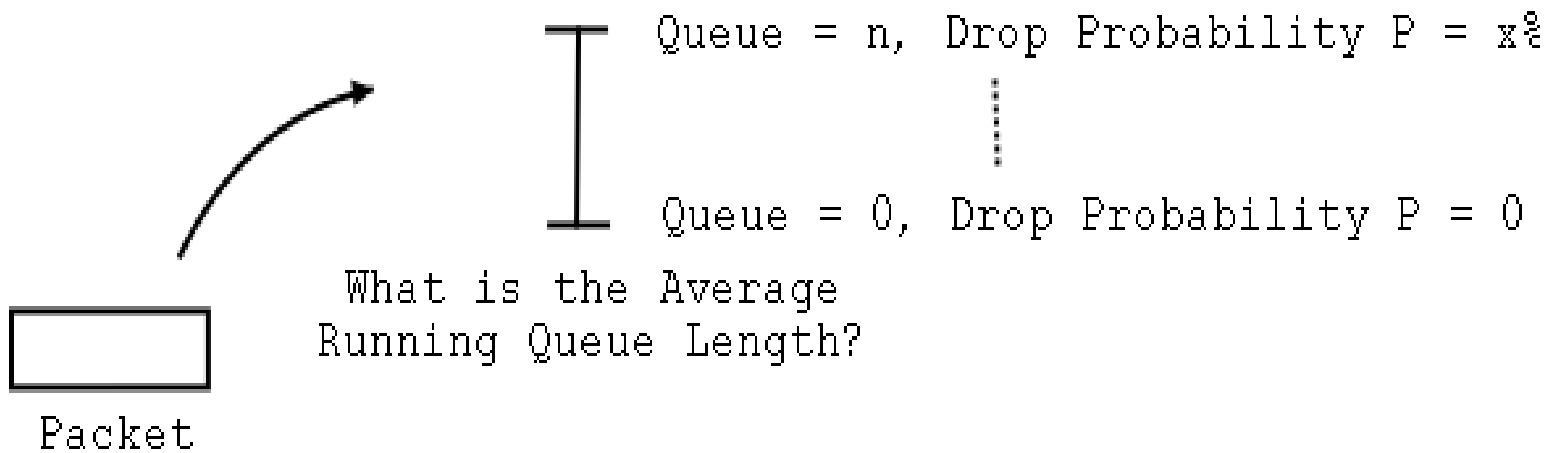
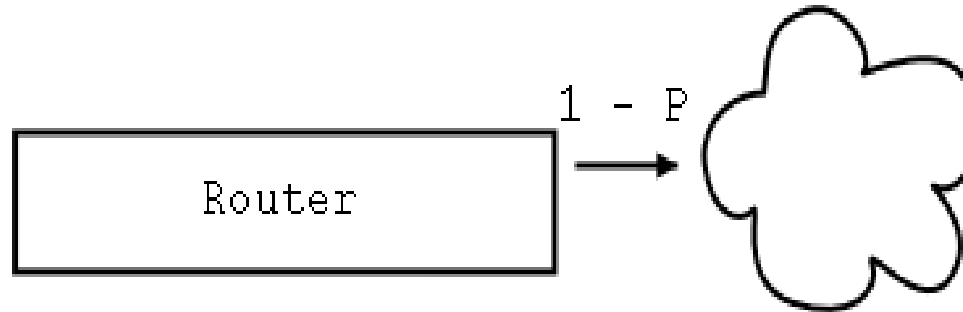
Active Queue Management

- One way to *control* congestion in the network
- Tail drop (FIFO queueing)
- Random Early Detection (RED)
- Explicit Congestion Notification (ECN)
 - Probably coupled with RED
- Ongoing experimentation and research
- Implementations available

Tail Drop Illustrated



RED Illustrated



Scheduling

- Alter transmission order of packets
- Can be based on:
 - IP Addresses
 - Priority (e.g. ToS bits)
 - Protocols (e.g. SSH)
 - Flow characteristics
- Must define capacity/weight for queues

Traffic Shaping

- TCP rate control
 - Alter TCP receiver window *on the fly*
- ACK pacing
 - Slow or spread out ACKs to control sender
- Packeteer (middlebox) does this
 - Yes, it can be a little scary
- Can be implemented in end host stacks

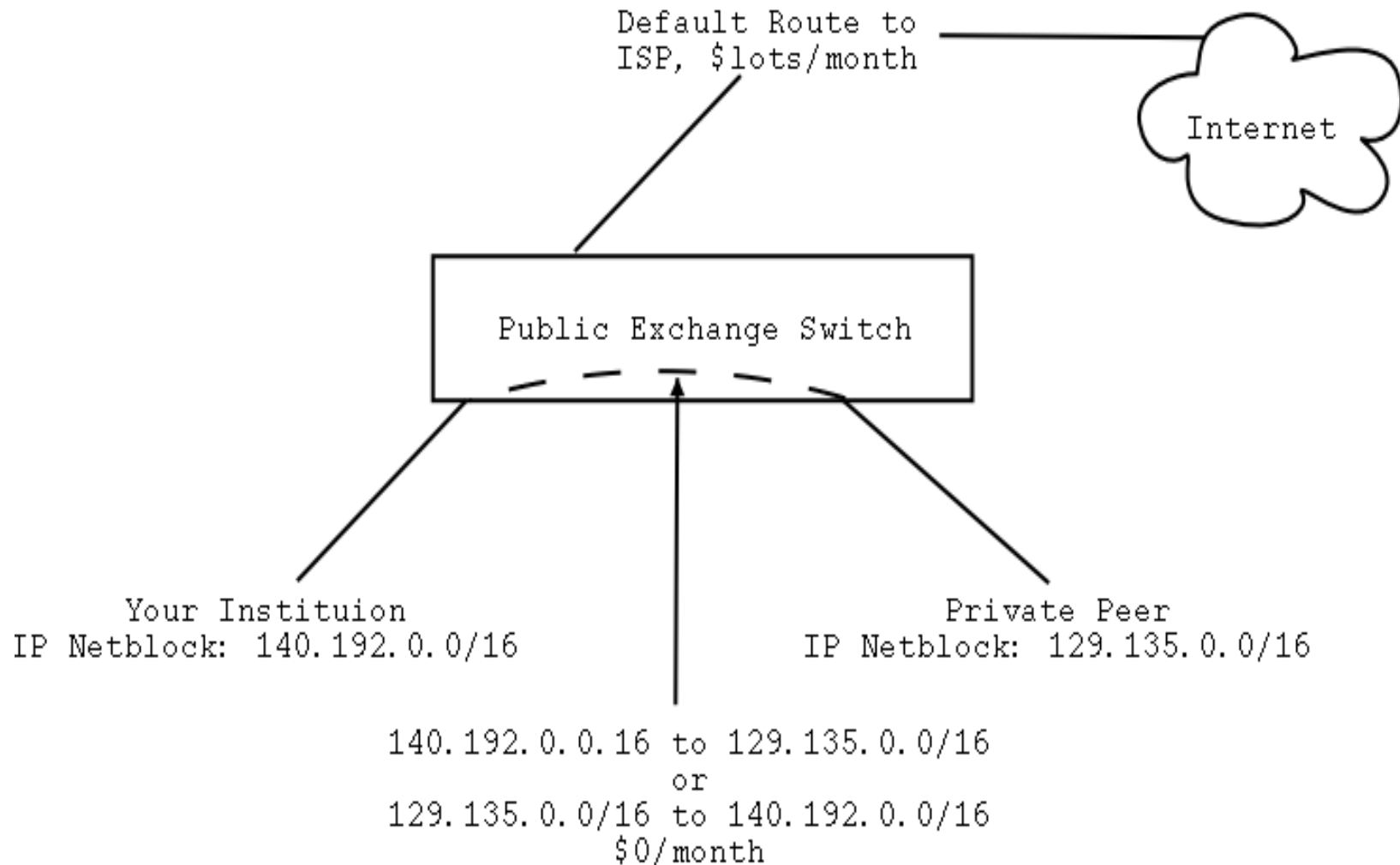
Caching

- Transparent
 - It is there, but users don't know it
- Voluntary
 - It is there, but users must know to use it
- Probably only buys a short amount of time

Private Peering

- There is such a thing as a free lunch!
- Well, OK not really
- Involves some routing complexity
- Startup cost might be high
- You may have a choice of transit provider
- If you can get to an exchange, do so!

Private Peering Illustrated



Monitoring

- Some tools:
 - MRTG, RRDTool, cflowd
- Some things to watch:
 - Queue depth
 - Packet drops
 - Link utilization
 - Buffer utilization
- Latency versus throughput

Consortia

- Some nice things in your own back yard?
- Might be free, low cost or subsidized
- May at least be lots of capacity
- Might also be worth what you pay
- Examples:
 - Internet2
 - Illinois Century Network
 - STAR TAP

Proxy Servers

- Lots of opportunity for control
- Can do lots of the capacity solutions at once
- Not sure that you want them to
- Lots of *middlebox* issues

Content Distribution

- Content providers move data closer to you
- Maybe setup up your own Red Hat mirror
- Akamai is well known in this space
- A form of load balancing

Content Subscription

- Obtain local copies of data for distribution
- Sort of like a library service
- You do not own the content
- May help alleviate copyright issues
- iBEAM is popular in this space

Network Address Translation (NAT)

- Intended as solution to IP address shortage
- Has a number of well documented problems
- Probably not your capacity solution
- Probably wouldn't help much anyway
- In fact, it would probably hurt you more
- Not recommended if you have addresses
- Bad Juju

What Would I Do? (Bias Slide)

- Oversubscription before CoS/QoS
- Preserve *End-to-end model*
- Get to an exchange and peer
- Get into a consortium like Internet2
- Do lots of monitoring (understand traffic)
- Be wary of *silicon snake oil*
- Be willing to research and test anything

References

<http://condor.depaul.edu/~jkristof/>

<http://www.aciri.org/floyd/>

<http://www.ietf.org>

<http://www.nanog.org>

<http://listserv.nd.edu/archives/resnet-1.html>

<http://www.theorygroup.com/Archive/Unisog/>

<http://darkwing.uoregon.edu/~joe/how-to-go-fast.ppt>