Distributed Denial of Service

John Ioannidis

ji@research.att.com

AT&T Labs – Research

Joint work with Steve Bellovin, Matt Blaze (AT&T),
Sally Floyd, Vern Paxson, Scott Shenker (ICIR),
Ratul Mahajan (University of Washington),
Angelos Keromytis (Columbia University)

Another Talk about DDoS?

- Fundamentals of DDoS.
- Taxonomy of attacks and defenses.
- Pushback: a promising solution.
- Limits of the technology.

Security Attacks

- Ordinary security attacks:
  - Exploit target vulnerabilities.
  - Give resources to attacker.
  - Fixing vulnerability solves the problem.

- Denial-of-service attacks:
  - Prevent others from using the targetted system.
  - Attacker gets indirect benefits.
  - Some DoS attacks exploit vulnerabilities.
  - Some just eat up resources.

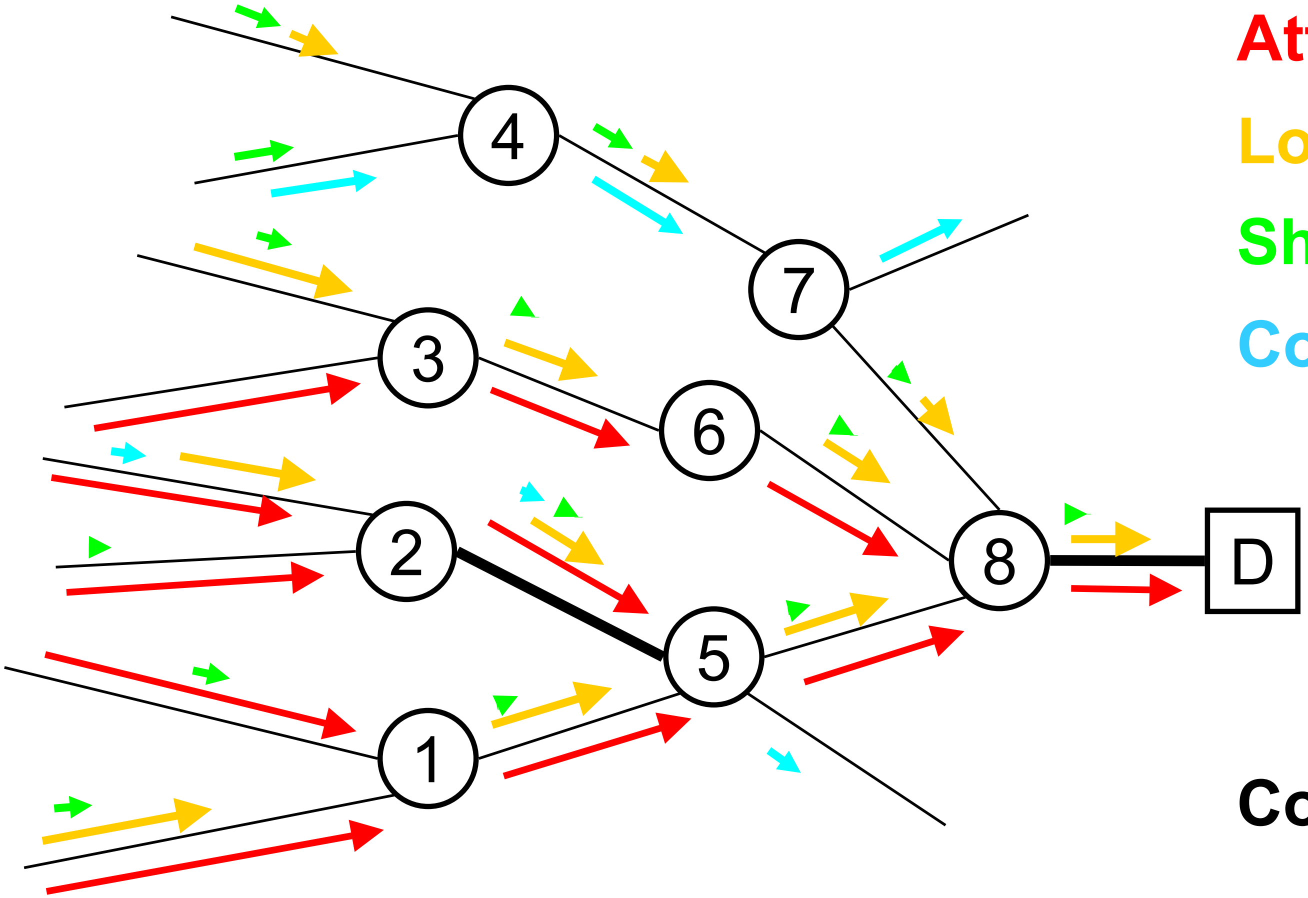
DDoS: Distributed Denial of Service

- Network attacks aim at flooding the victim's link.
  - Strong attacker against weak victim.

- Distributed attacks use multiple sources.
  - Lots of weak attackers against strong victim.

- Usual implementation:
  - Multiple compromised machines with *zombies*.
  - Masters direct zombies to attack victim.
  - Victim overwhelmed.

An Attack in Progress



Attack

Looks like attack

Shares links

Collateral damage

Congested link

During the Attack

- Bad (attack) traffic does not obey E2ECC, floods links.
- Poor (looks like attack) traffic obeys E2ECC, backs off due to congestion on links 2-5 and 8-D.
- As does good (shares links with attack) traffic.
- Some unrelated (collateral damage) traffic backs off due to congestion on the 2-5 link.

Why are DDoS Attacks Hard to Defend Against?

- Nothing victims can do to protect themselves:
  - Usually, attack is on *connectivity*.
  - Better overall host security *would* help.
  - As would source address filtering.
- Bandwidth management not applicable.
  - End-to-end congestion control not applicable.
    - Source does not obey E2E CC.
  - Active Queue Management not applicable.
    - ‘Flows’ very short-lived.
- Diffserv/Intserv do not help with best-effort traffic reqs.

A Taxonomy for DDoS

- Attack
  - End node
  - Network
- Defense
  - Detection of attack
  - Response
    - Traffic management
    - Scope
    - Timeliness
    - Extent
    - Impact

Attacks I: Target is end node.

- CPU attack.
  - Cause unnecessary crypto to happen.
- Memory attack.
  - SYN flood.

Easier to mitigate:

- Proper host security.
- Proper protocol design (*e.g.,* cookies).

Attacks II: Target is network link

- By target link:
  - Usually access link (core network over-provisioned).
  - Slow border links (to distant lands).
- By source of attack:
  - Trin00/TFN style: master/slaves.
  - Reflector attacks.
- By packet contents:
  - Random (just bandwidth).
  - Calculated (SYN, directed broadcasts).
- By cause:
  - Deliberate attack.
  - Flash crowd.

Characteristics of Current Attacks

- Several small-scale attacks a day.
- Fairly crude.
  - Captured code is of abysmal quality.
  - Have not paired up with virus writers.
- Anisotropic.
  - Locality of penetrated machines.
  - Internet too large to design an isotropic attack.
- Purpose seems to be ego gratification.
  - IRC turf wars.
  - Vandalism.
  - We worry about tactical uses.

Design of a Perfect Attack

- Looks like legitimate traffic.
  - Flash crowd.
- Isotropic/topology aware.
  - Uses network mapping information.
- Adaptive.
  - Responds to our attempts to quench it.
- Automatic propagation.
  - Viruses or other software flaws.

Why is the network still running?

Defense I: Detection

- Passive *vs.* active:
  - Traffic monitoring.
    - Content.
    - Shape/characteristics.
  - Traffic marking.
    - ICMP TRACEBACK
    - Packet marking (many variants).
- Distribution:
  - Single point.
  - Collaborative.

Defense I: Detection, cont’d

- Timing:
  - Proactive (runs at all times).
  - Reactive (turned on in response to attack).
- Feedback:
  - From response mechanisms.

Defense II: Response

- Traffic management:
  - Rate limiting.
  - Filtering.
  - Redirection.
- Distribution:
  - At specific points.
  - Along attack path.
- Scope:
  - Within an administrative domain (intra-ISP).
  - Across administrative domains.

Defense II: Response, cont’d

- Cost/benefit:
  - Innocent traffic.
  - Collateral damage.
  - Level of service.
  - Vulnerabilities introduced?
- Identification of attacker?
  - Just mitigating effects.
  - Finding attacker and/or zombies.
    - Punishing.
    - Fixing!

Limits

- Described an N-space of attacks and defenses.
- Need to identify limits of solution space.
- Need to define metrics of success.
  - Limiting case: flash crowd.
- Unexplored dimensions?
  - Legal/social.
  - Radically change Internet architecture.
    - Per-packet payment?
    - Back to virtual circuits?
    - Active networks?

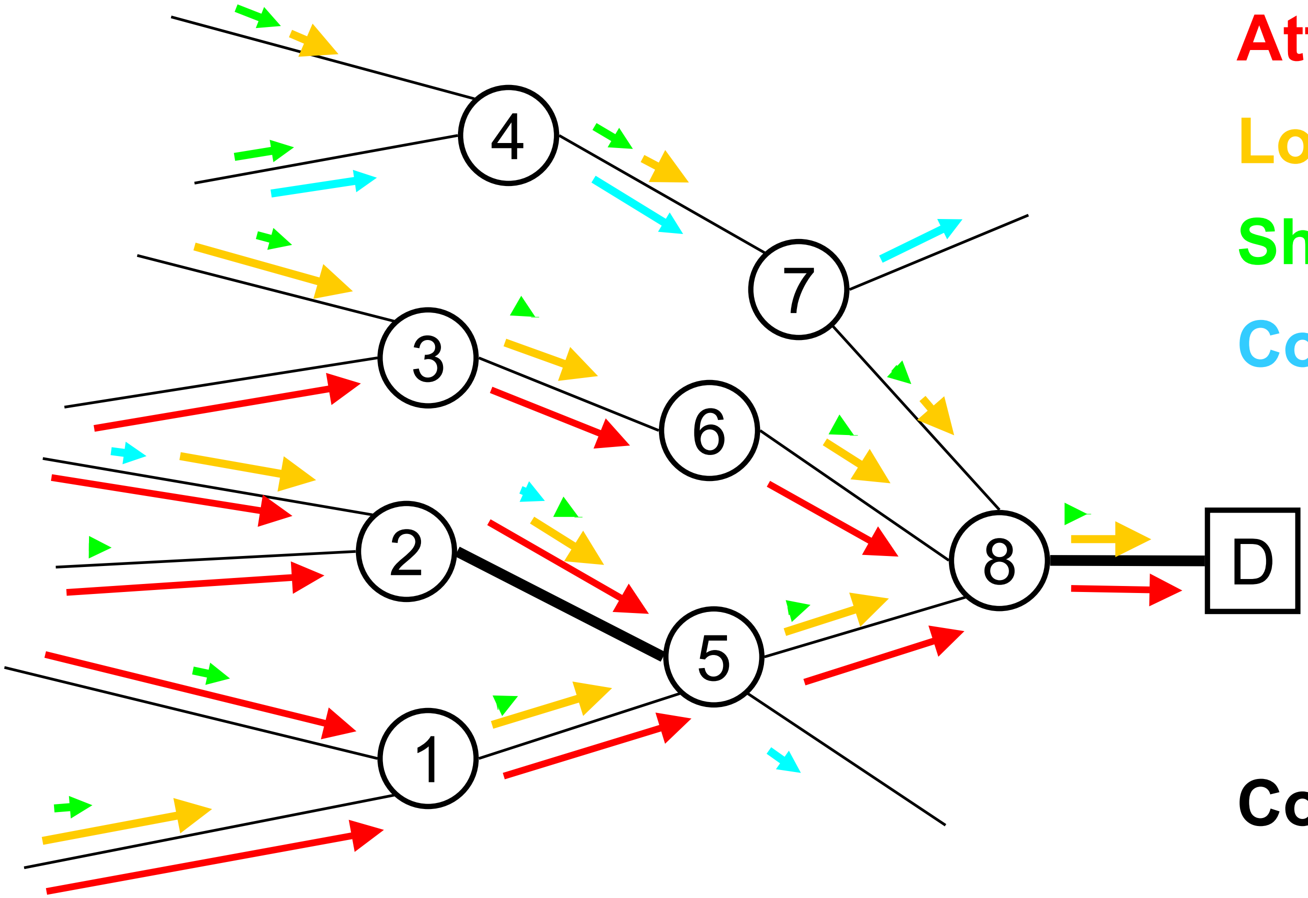
Pushback

- Router-based solution against bandwidth attacks.
- Attack: too many packet drops on a particular link.
- Aggregate: set of packets with a common feature.
- Find the most common aggregate in the drop set.
- Aggregate-based Congestion Control (ACC).
- Local ACC (LACC):
  - Works independently on congested links.
- Pushback:
  - Tell upstream routers to also drop aggregate.

Involving the Routers

- Detect.
  - Which routers detect the attack?
  - How do they do it?
- Inform.
  - Are other routers informed?
  - How?
- Limit.
  - What traffic is dropped?
  - Where?

An Attack in Progress (again)



Attack

Looks like attack

Shares links

Collateral damage

Congested link

Local Rate-limiting

- Router 8 can preferentially drop bad traffic.
- This would allow more good traffic to flow in from 7…
- but would not improve things for the rest.
  - Too much traffic coming in from 5 and 6.
  - **Collateral damage** is still occurring.

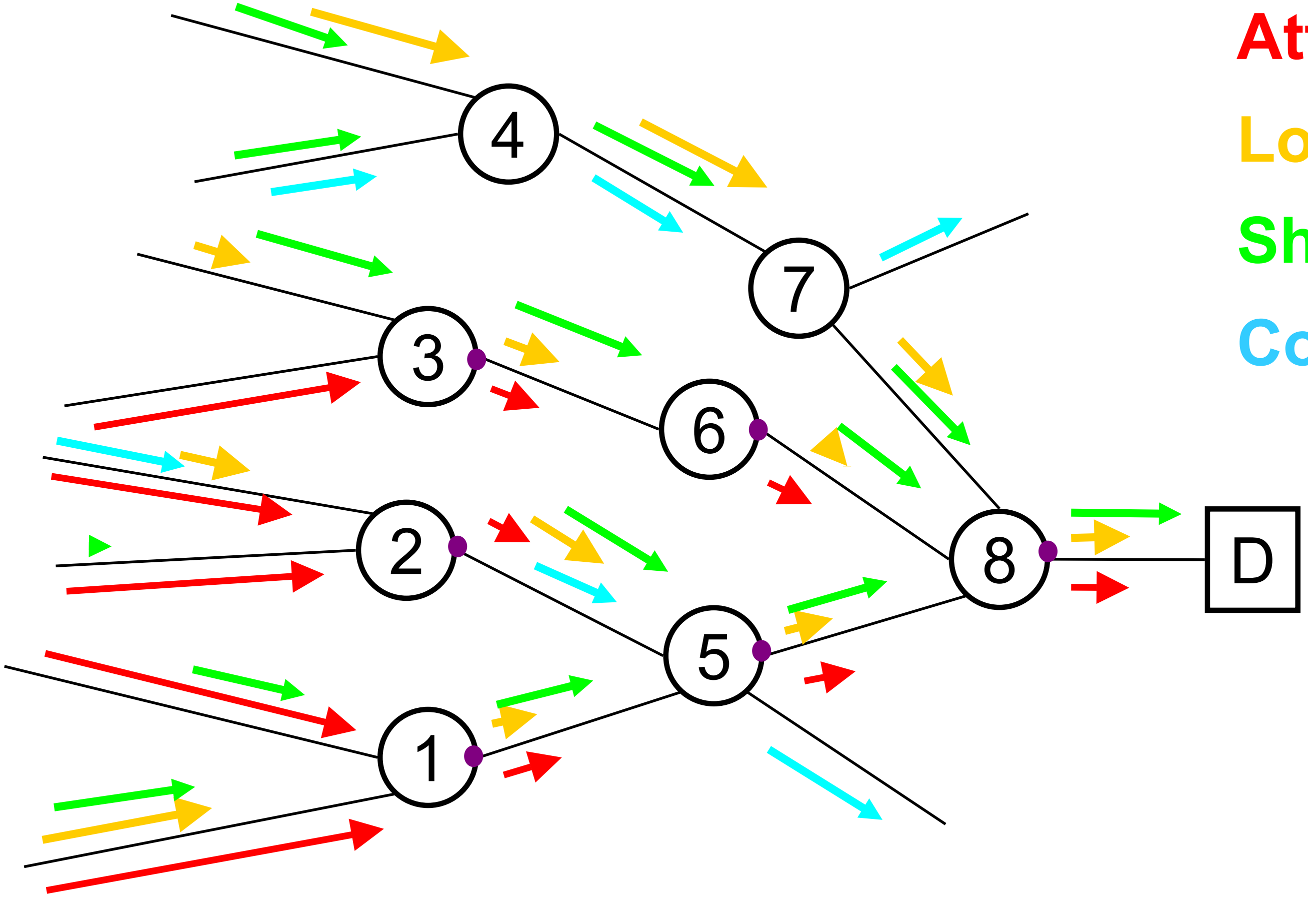
Pushback

- Router 8 rate-limits.
- Detects where bad traffic is coming from.
- Directs upstream routers where most of traffic is coming from (5 and 6) to rate-limit on its behalf.

- Recurse!

- 5 and 6 now “see” congestion for bad traffic and push further back.
- More non-attack traffic flows.

After Pushback



Attack

Looks like attack

Shares links

Collateral damage

Involving the Routers

- Detect.
  - Which routers detect the attack?
  - How do they do it?
- Inform.
  - Are other routers informed?
  - How?
- Limit.
  - What traffic is dropped?
  - Where?

Setting the ‘Evil Bit’

- Attack signature and congestion signature.
- CS approximates the Evil Bit.
- Detect what the Aggregate should be.

- Collect set of dropped packets (‘drop set’).
  - Just sampling is OK.
- Match against forwarding table.
  - Easy way of identifying attacks on subnets.
- Pick most frequent prefix(es).
- Update the rate limiter.

Involving the Routers

- Detect.
  - Which routers detect the attack?
  - How do they do it?
- Inform.
  - Are other routers informed?
  - How?
- Limit.
  - What traffic is dropped?
  - Where?

Pushback Messages

- If local rate-limiting decreases congestion, stop.
- Otherwise, tell upstream routers with the largest contributions to also rate-limit (pushback request).
- Upstream routers apply the same algorithm to decide whether to propagate (push further back).
- Originator sets a maximum depth.
- Downstream messages provide feedback.

- No explicit acknowledgements or crash recovery (soft state).

Involving the Routers

- Detect.
  - Which routers detect the attack?
  - How do they do it?
- Inform.
  - Are other routers informed?
  - How?
- Limit.
  - What traffic is dropped?
  - Where?

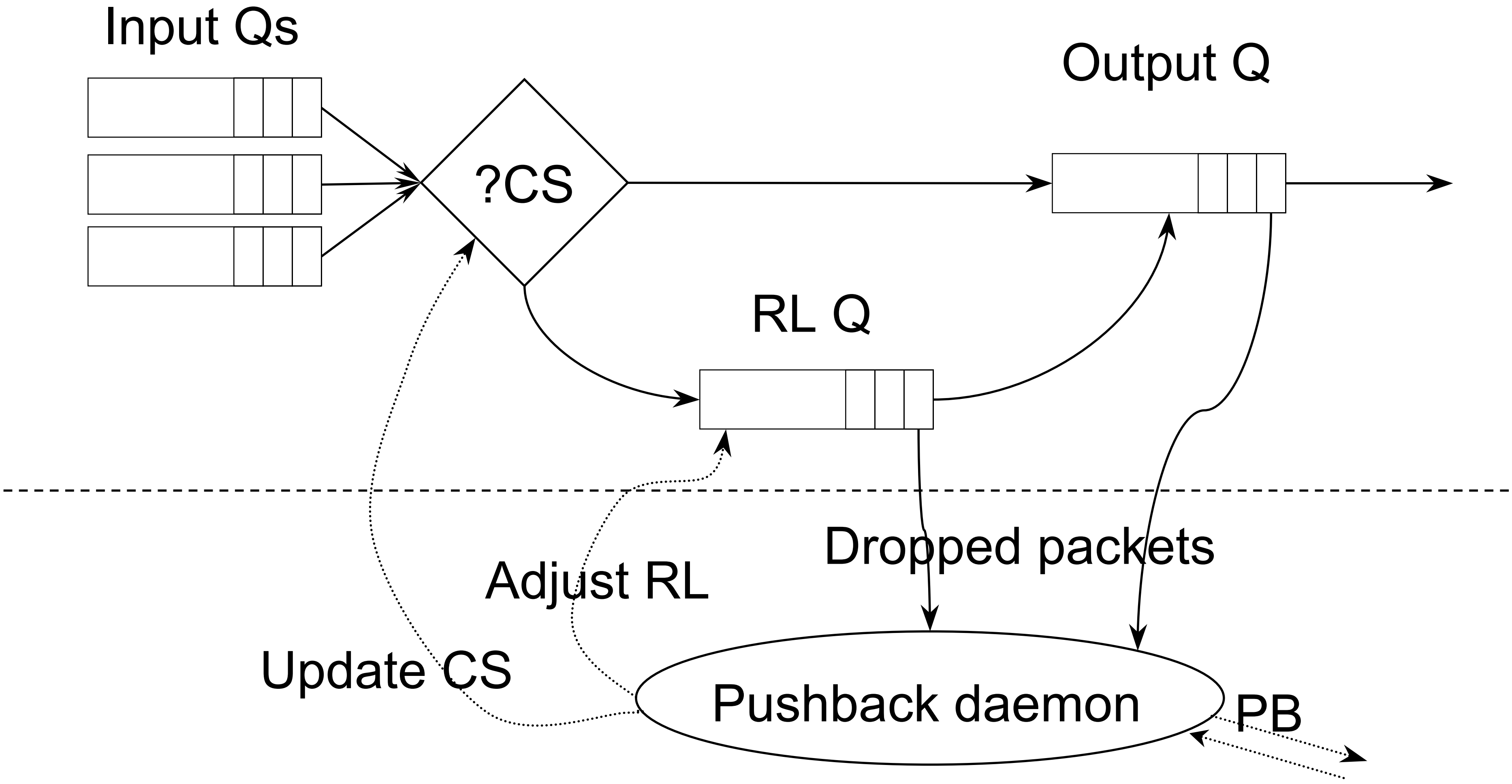
Local Rate Limiting

- Source can be local or downstream `pushbackd`.
- Preferentially drop packets matching aggregate(s).

- Implemented as IPFW ‘`pipe`’under FreeBSD.
- Will also be done with ALTQ.
- Commercial routers have various ways of rate-limiting.

- Packets admitted by the RL passed to Output Q.
  - No preferential treatment!

Pushback Router Architecture



Decoupling of Components

- `pushbackd` need not run on the routers.
  - Router can sample the drop set
  - Attached processor updates router’s RL.
- Different machines can run different detection algorithms.
- But we must agree on format of pushback messages.
  - `draft-floyd-pushback-messages-0?.txt`

Knobs to Adjust

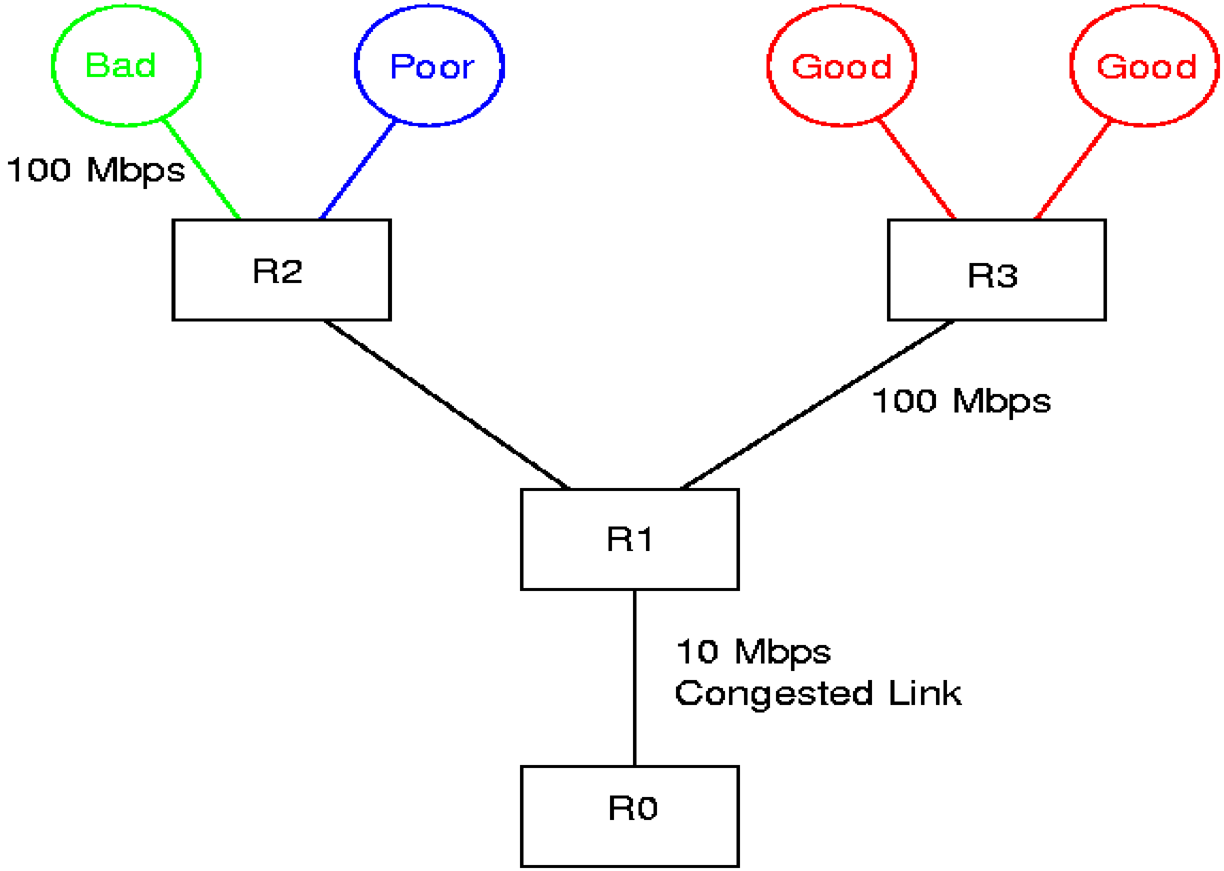
- Congestion signature derivation.

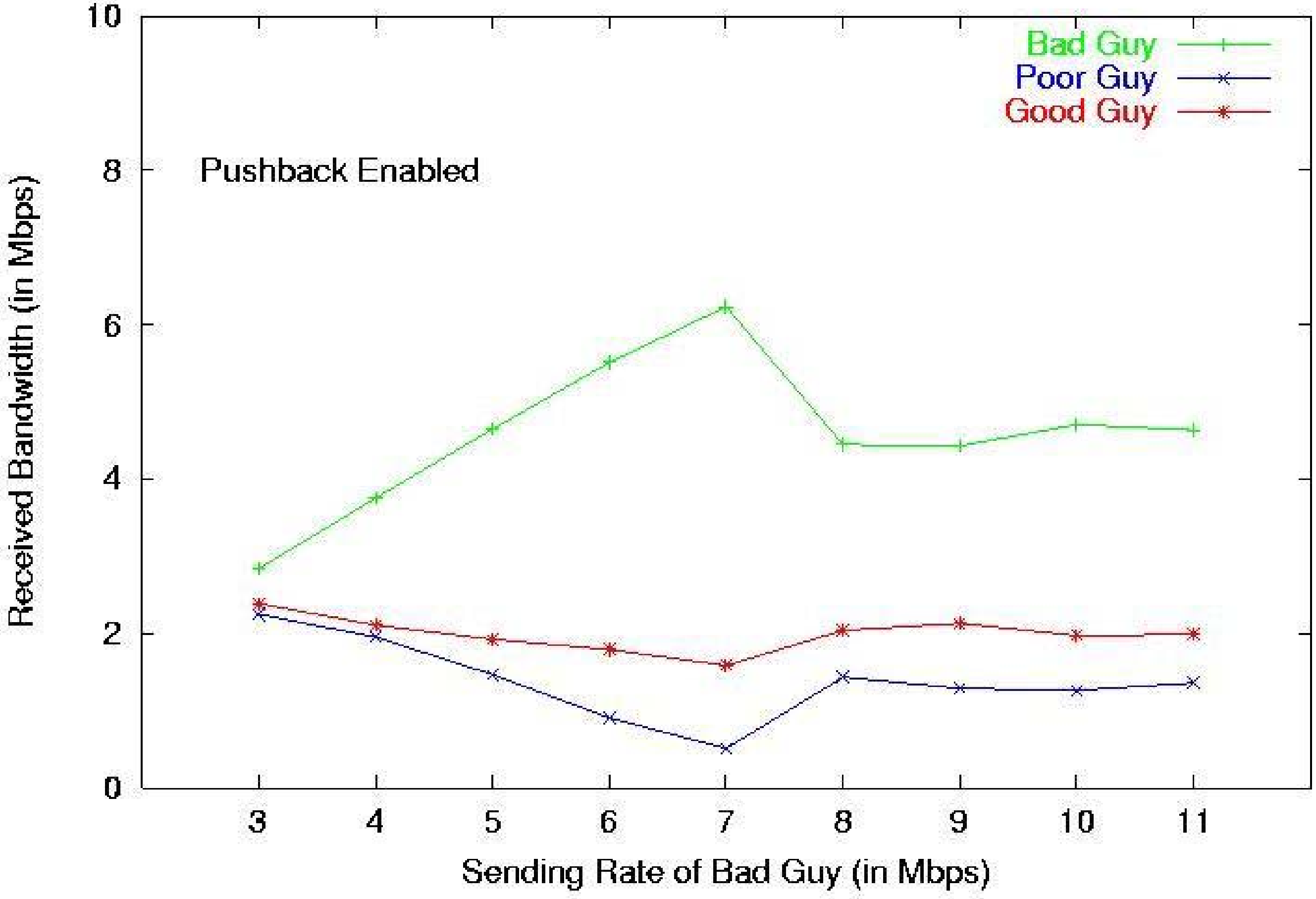
- Rate-limiting aggressiveness.
- Upstream distribution of bandwidth rate-limiting requests.
- Damping of feedback control loop
  - (between requests for limiting and results).

Simulation Results

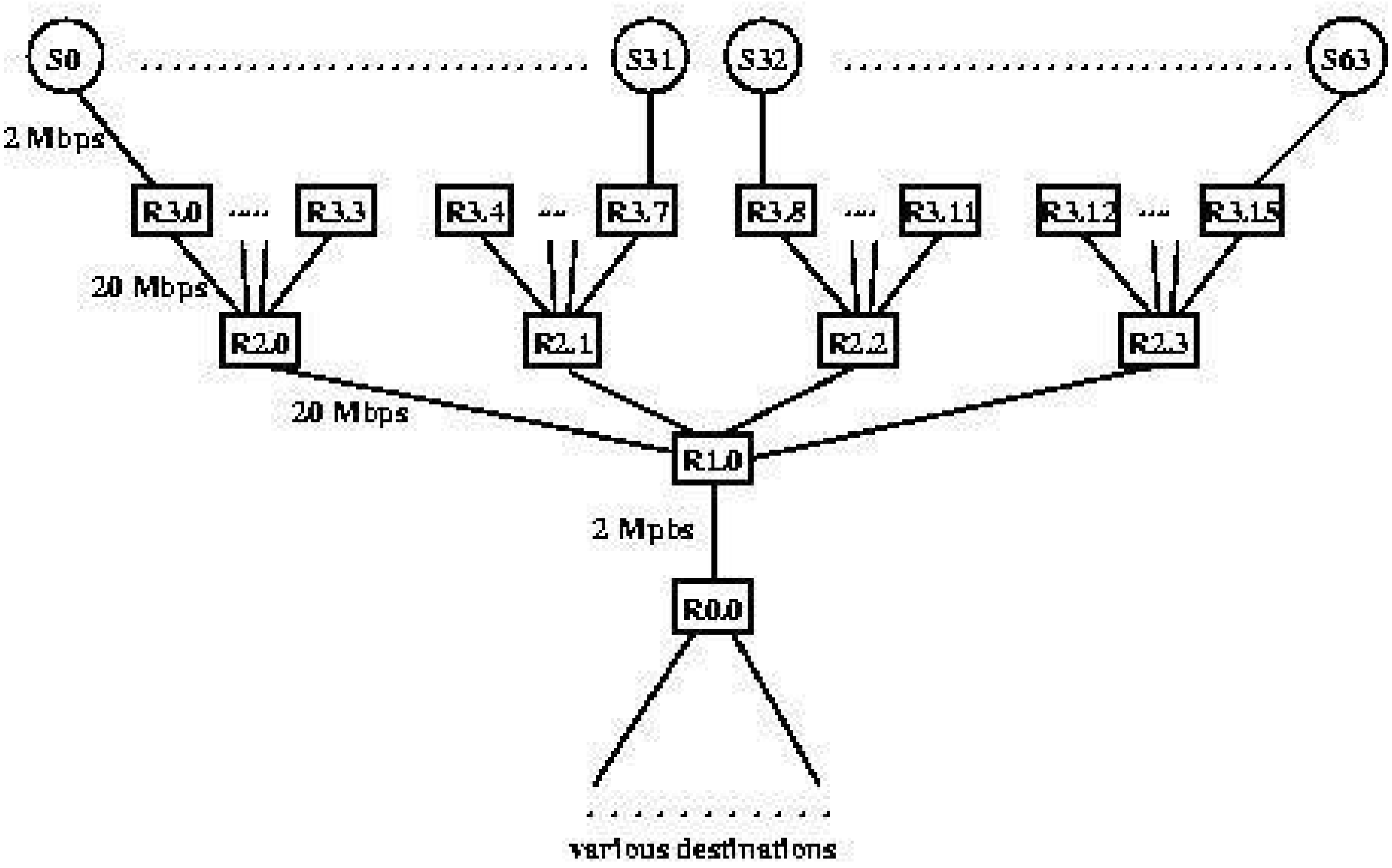
- Various topologies.
  - Small attacks.
  - Large attacks.
  - Flash crowds.
- Various bandwidth allocation algorithms.
- Various detection algorithms.

Small Topology

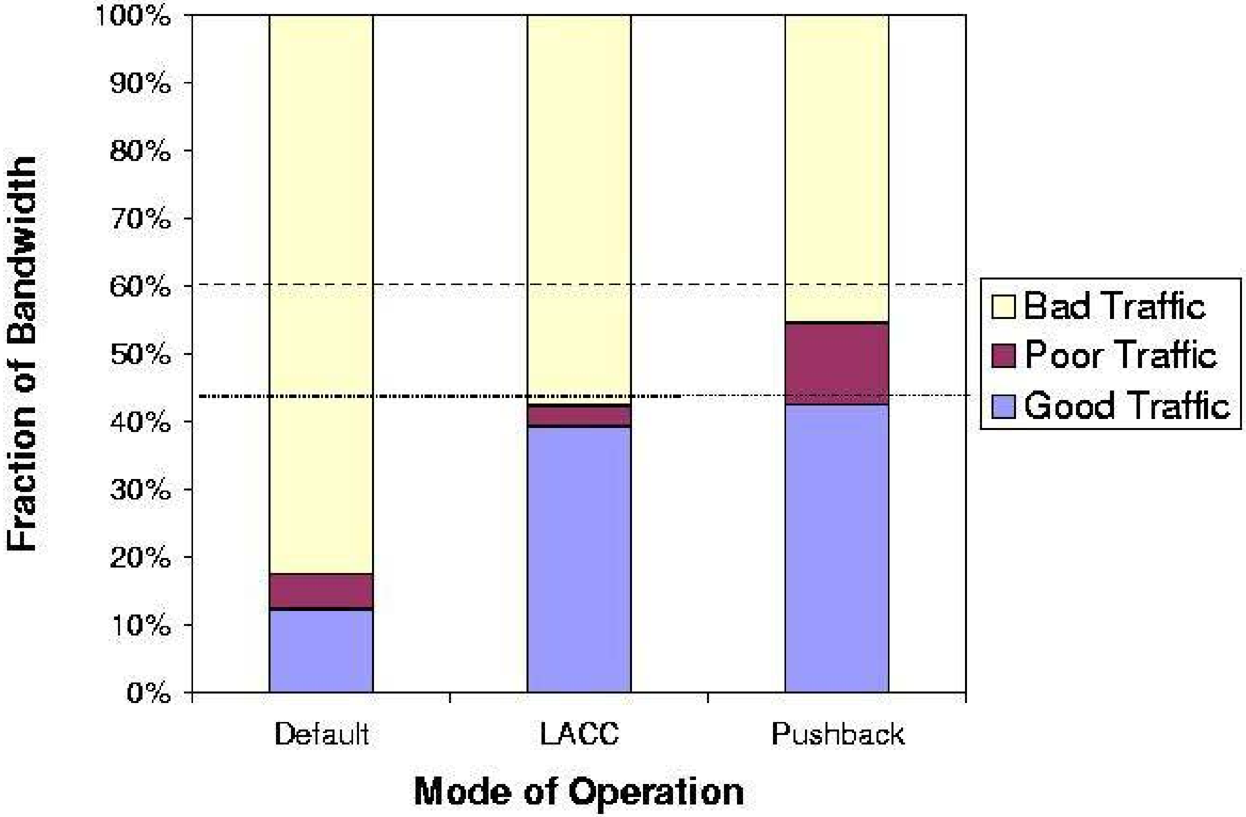



10
8
6
4
2
0
Received Bandwidth (in Mbps)
3
4
5
6
7
8
9
10
11
Sending Rate of Bad Guy (in Mbps)
Pushback Enabled
Bad Guy
Poor Guy
Good Guy

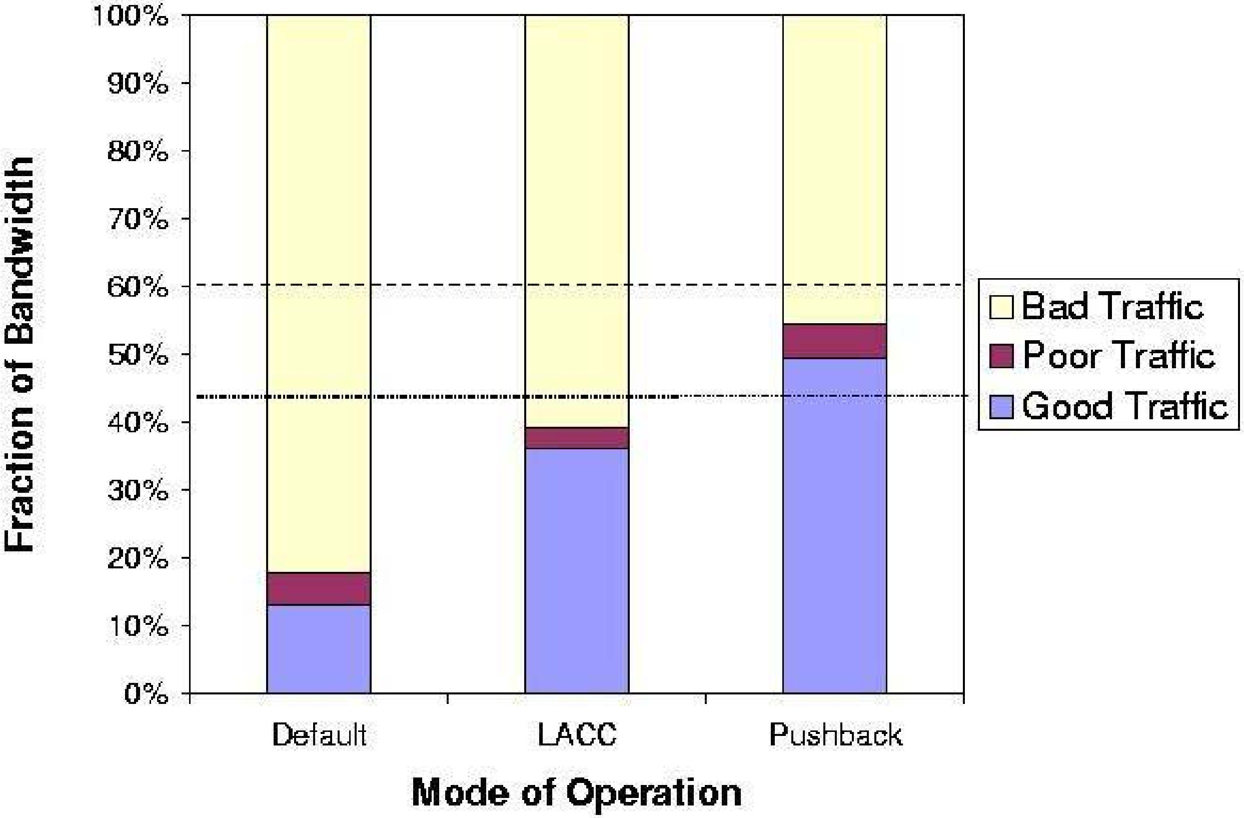
Big Topology, Web-like Behavior



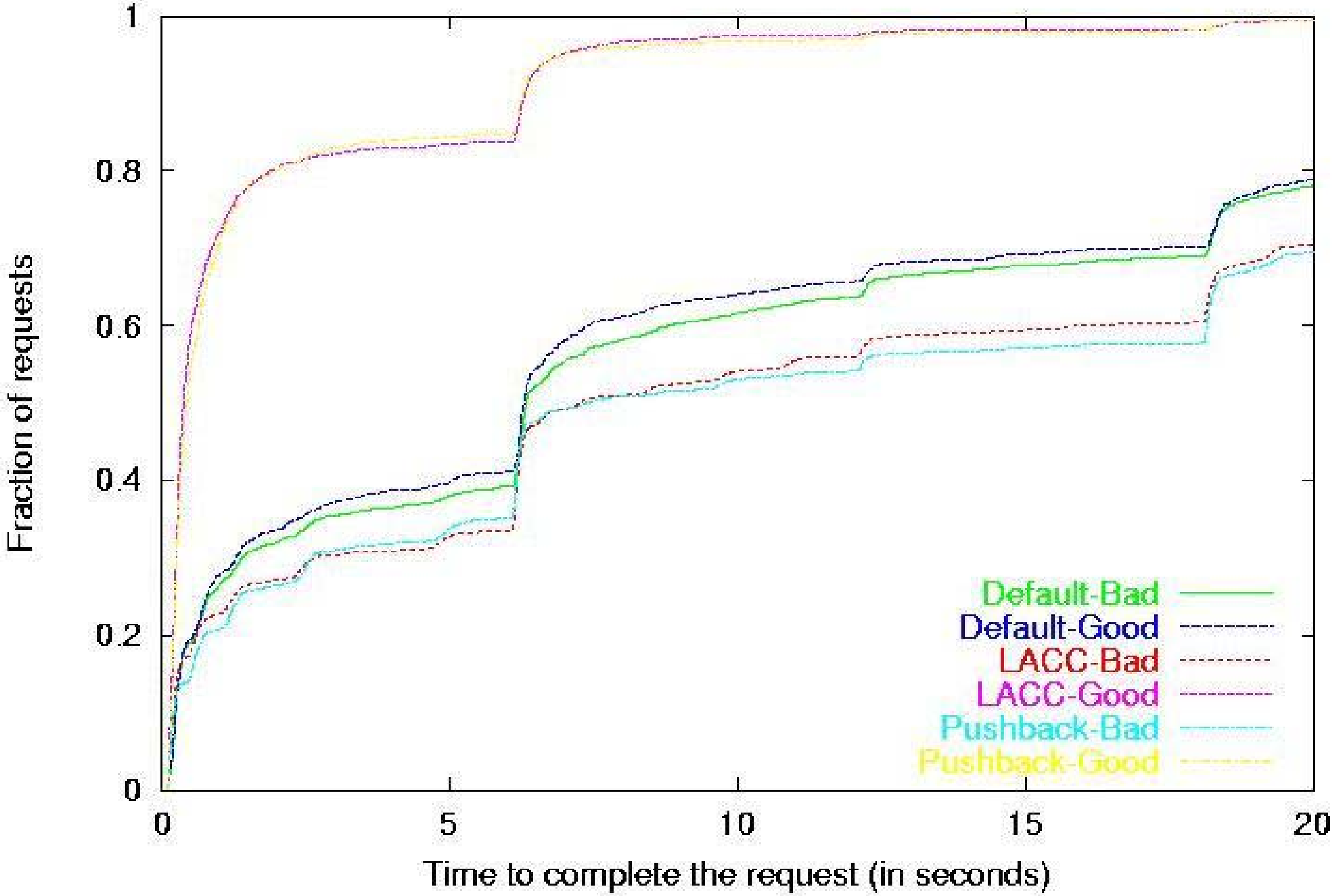
Throughput with 4 Bad Hosts



Throughput with 32 Bad Hosts



Flash Crowd Behavior



Problems

- Does not work for isotropic attacks.
  - Not an issue (yet).
- Flash crowds.
  - Penalizes traffic coming from fatter pipes.
  - May need to inject artificial asymmetry.
- Security.
  - TTL 255 for adjacent routers!
  - IPsec within same ISP.
  - Trust/policy issues at borders.
  - Should not become source of new abuses!

Future Work

- Better detection.
  - Traffic patterns.
  - Distributed monitoring.
  - IDS.
- Integration with commercial routers.
- Deployment in the field.
- Standardization efforts.

Summary

- DDoS treated as a congestion control problem.
- Handled inside the network.
- Three parts:
  - Detection.
    - Approximation of the evil bit.
  - Rate limiting.
    - Aggregate-based Congestion Control.
  - Pushback.
    - Involve upstream routers.

http://www.icir.org/pushback