Denying Denial of Service: A Tricky But Necessary Business

Stefan Savage

Chief Scientist, Asta Networks

Assistant Professor, UCSD

September 13, 2001

In the Beginning

Internet users worked together in harmony

Internet users, circa 1969

Times Have Clearly Changed

Overview

- How do DoS attacks work?
- How big a problem are they?

- Recent advances in DoS attacks
- What can be done: DoS defense methods

How do DoS attacks work?

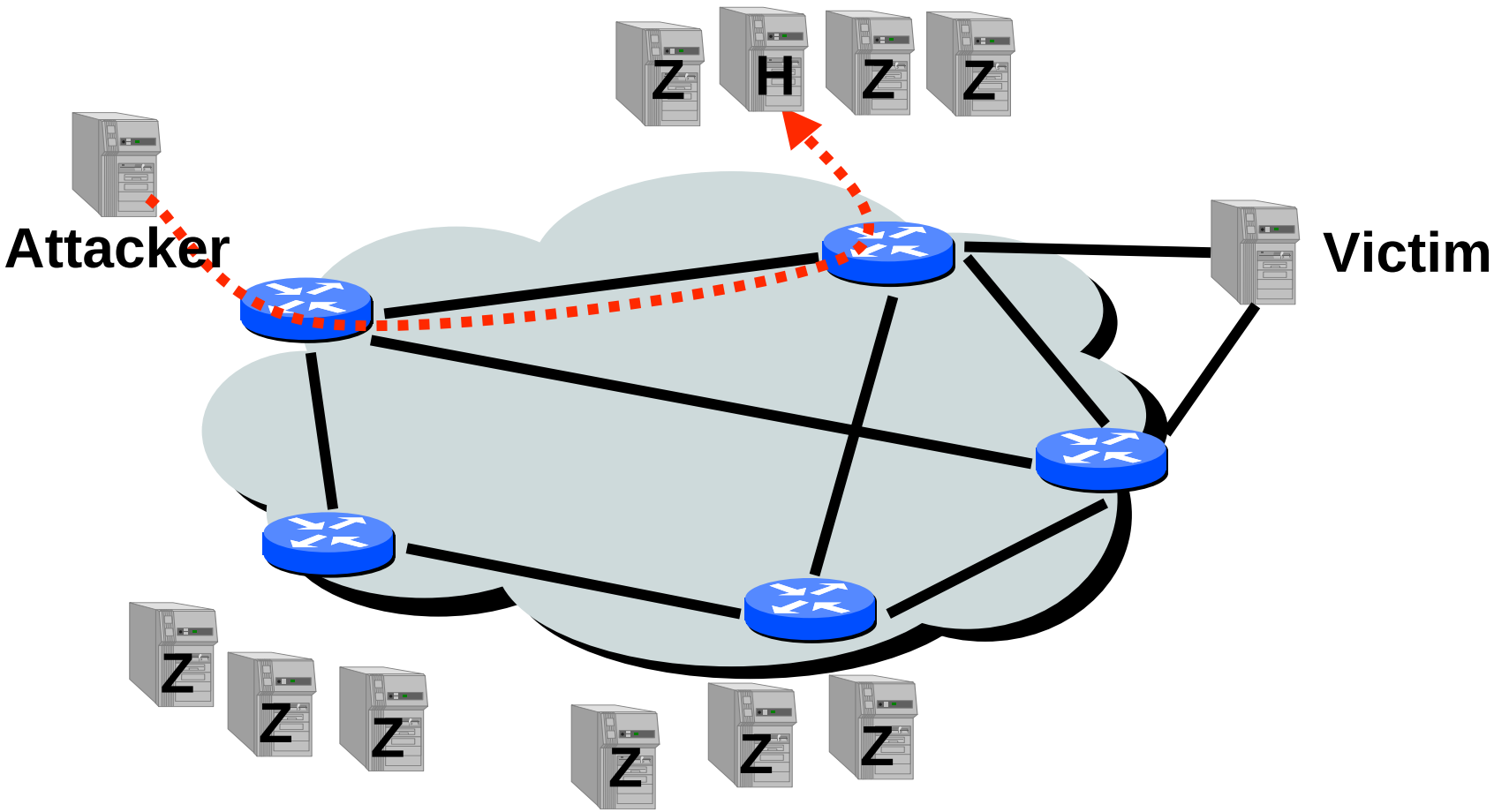
- Denial-of-Service attacks
  - **Logic:** exploit bugs to cause crash
    - e.g. Ping-of-Death, Land
  - **Flooding**: overwhelm with spurious requests
    - e.g. SYN flood, Smurf
- **Distributed** Denial-of-Service attacks
  - Flooding attack from multiple machines
  - More potent and harder to defend against

Step 1: Attacker infiltrates machines

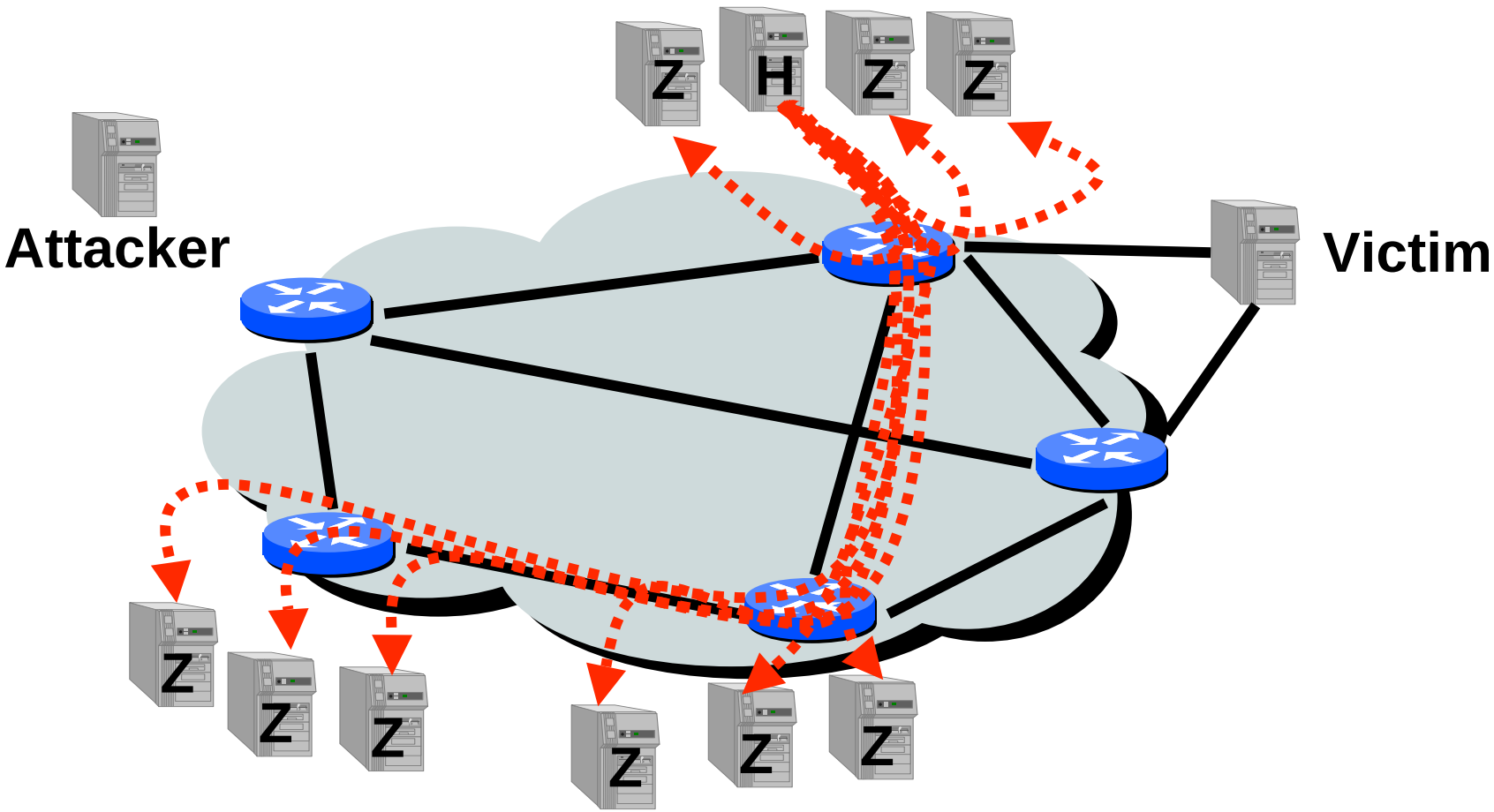
- Scan machines via Internet
- Exploit known bugs and vulnerabilities
- Install backdoor software
  - Zombie software (for attacking target)
  - Handler software (for controlling zombies)

- Cover tracks (e.g. **rootkit**)
- Repeat… (**highly automated**)

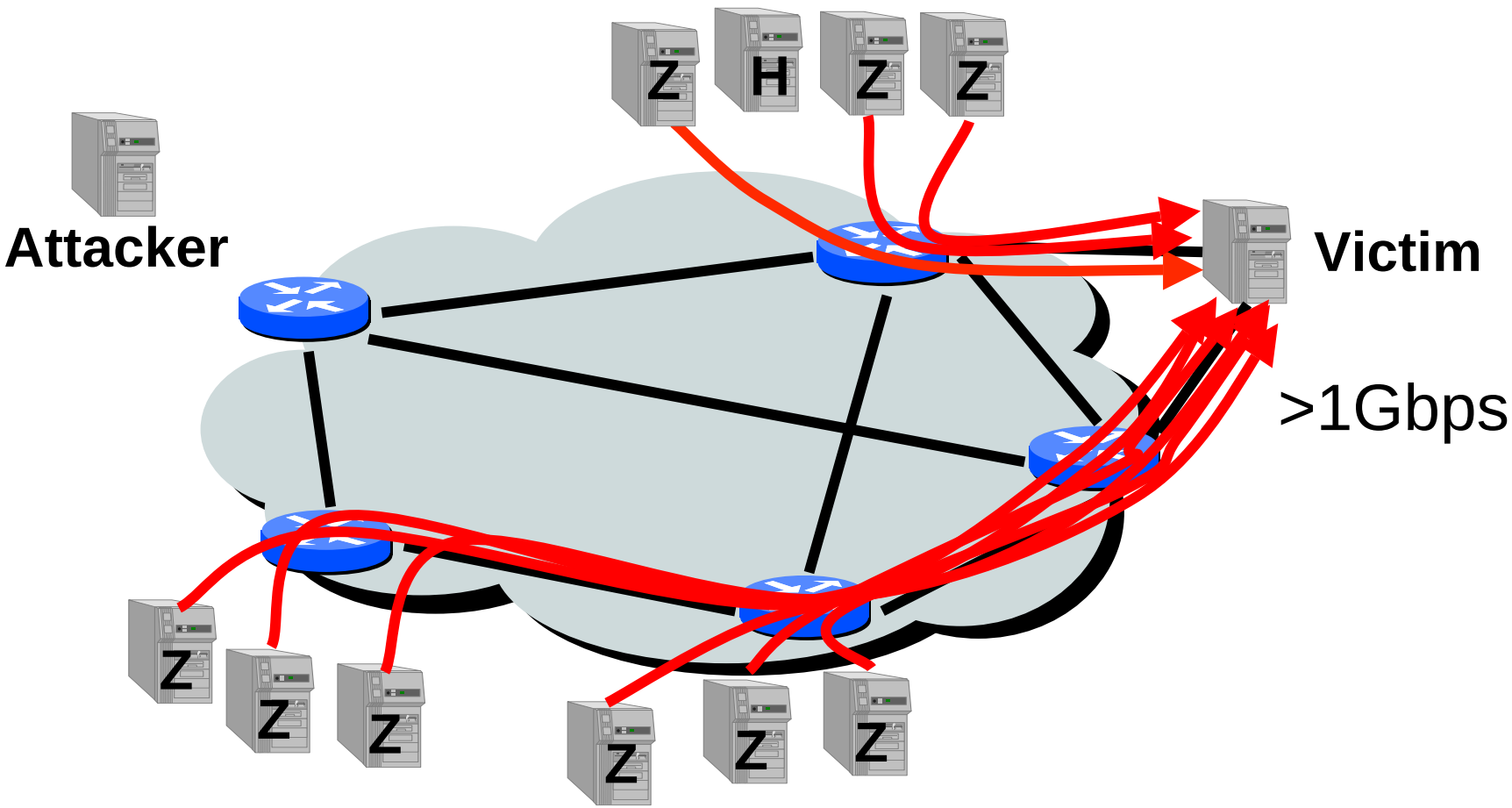
Step 2: Attacker sends commands to handler



Step 3: Handler sends commands to zombies



Step 4: Zombies attack target



Step 5: Victim suffers

- **Server CPU/Memory resources**
  - Consumes connection state (e.g. SYN flood)
  - Time to evaluate messages (interrupt livelock)
    - Some messages take “slow path” (e.g. invalid ACK)
  - Can cause new connections to be dropped and existing connections to time-out
- **Network resources**
  - Routers PPS limited, FIFO queuing
    - If attack is greater than forwarding capacity, good data will be dropped
    - Large attacks will disrupt BGP peering sessions
  - Attacks directly on router (e.g. ttl expire, target interfaces)
  - Random attacks across subnet can produce ARP storm

How big a problem is DoS?

- Traditional answer: "*Hard to say*"
  - A few highly publicized attacks
  - 2001 CSI/FBI survey says DoS reported by 38%
  - Until recently, no hard quantitative data available

- 2001 UCSD/CAIDA study: >**4000** attacks/wk
  - First measurement study of global DoS activity
  - New technique: *backscatter analysis*
  - Full paper appeared at USENIX Security '01:
    Moore,Voelker,Savage,
    "*Inferring Internet Denial of Service Activity*"

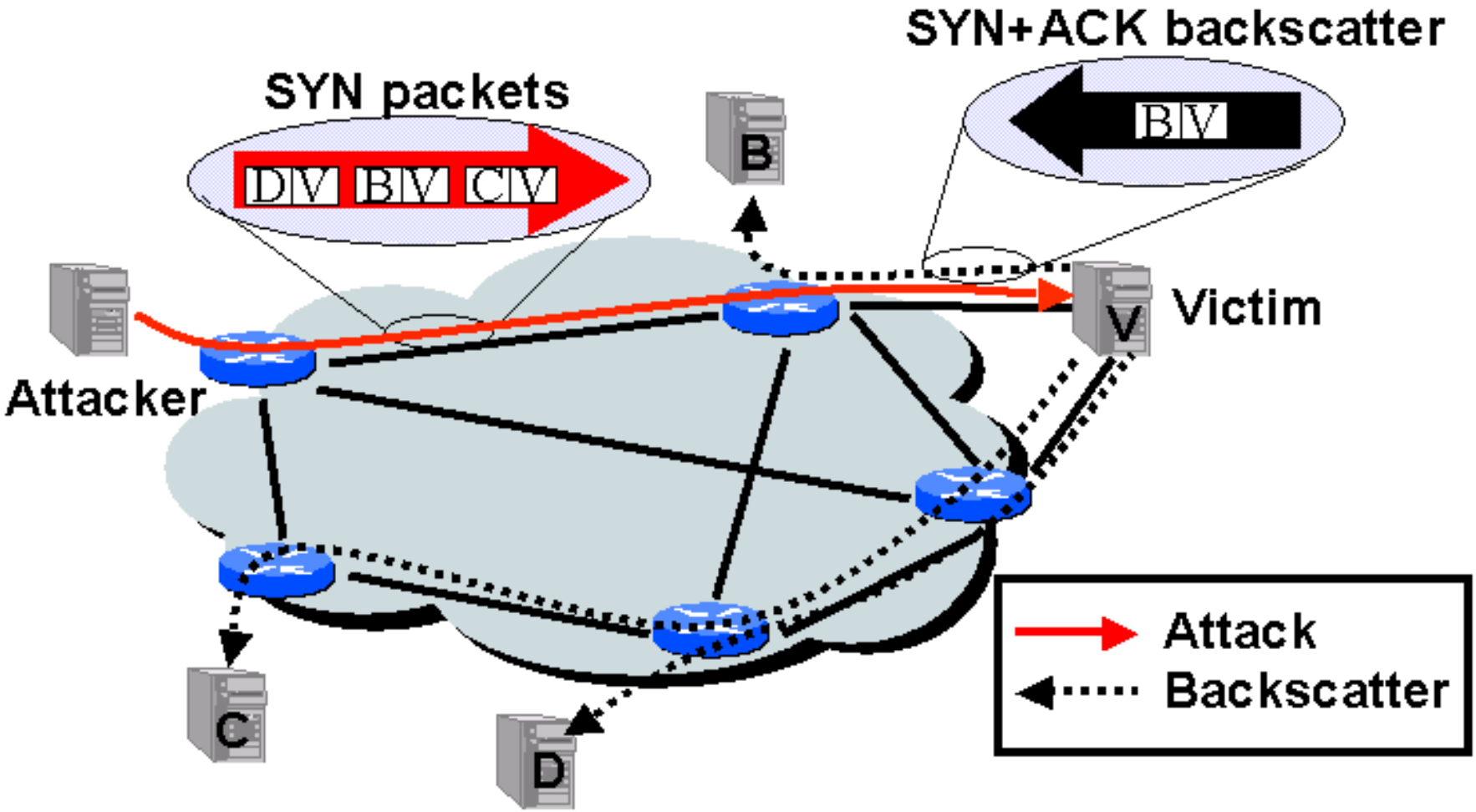
Backscatter analysis

- Key observations
  - Attackers “spoof” their source IP address randomly
  - Victims respond to these spoofed packets
  - Unsolicited responses (“*backscatter*”) are therefore equi-probably distributed around the Internet

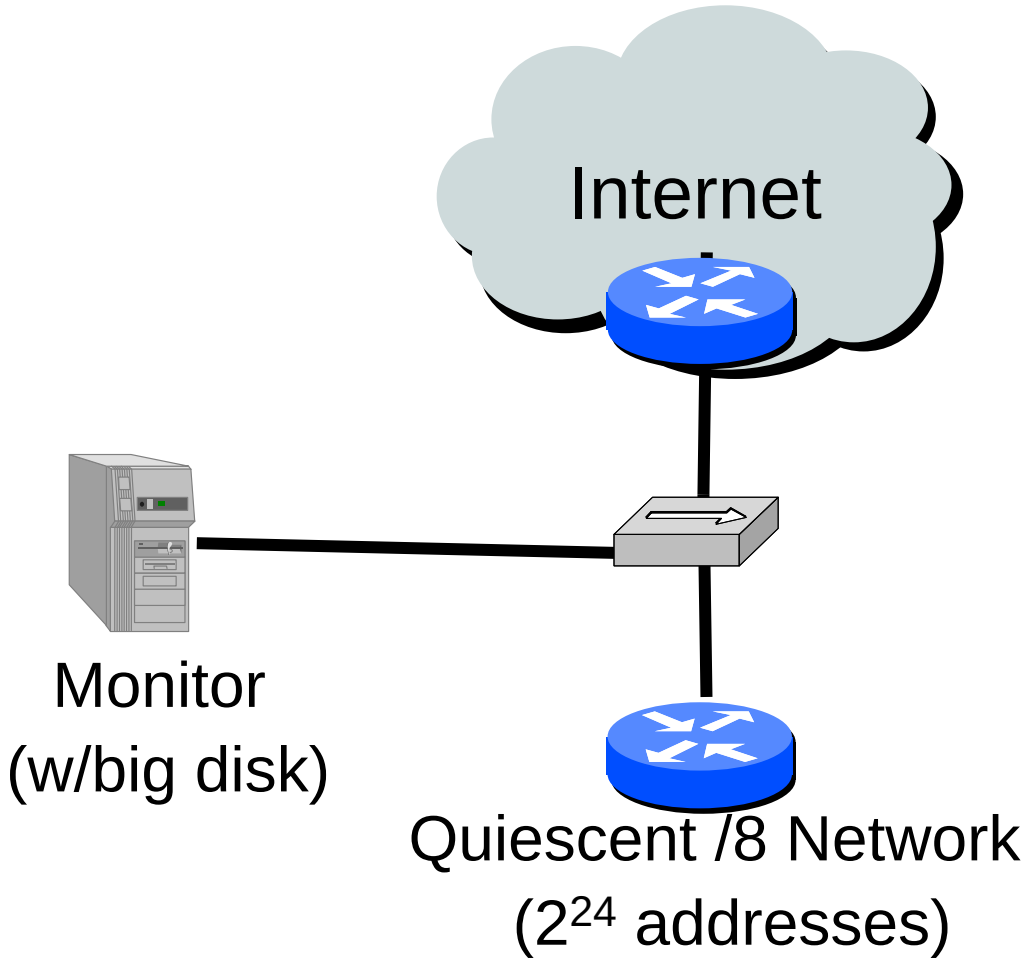
- Approach
  - Infer attacks by sampling block of *n* IP addresses
  - Expected backscatter packets for attack of *m* packets:

$$E(X) = \frac{nm}{2^{32}}$$

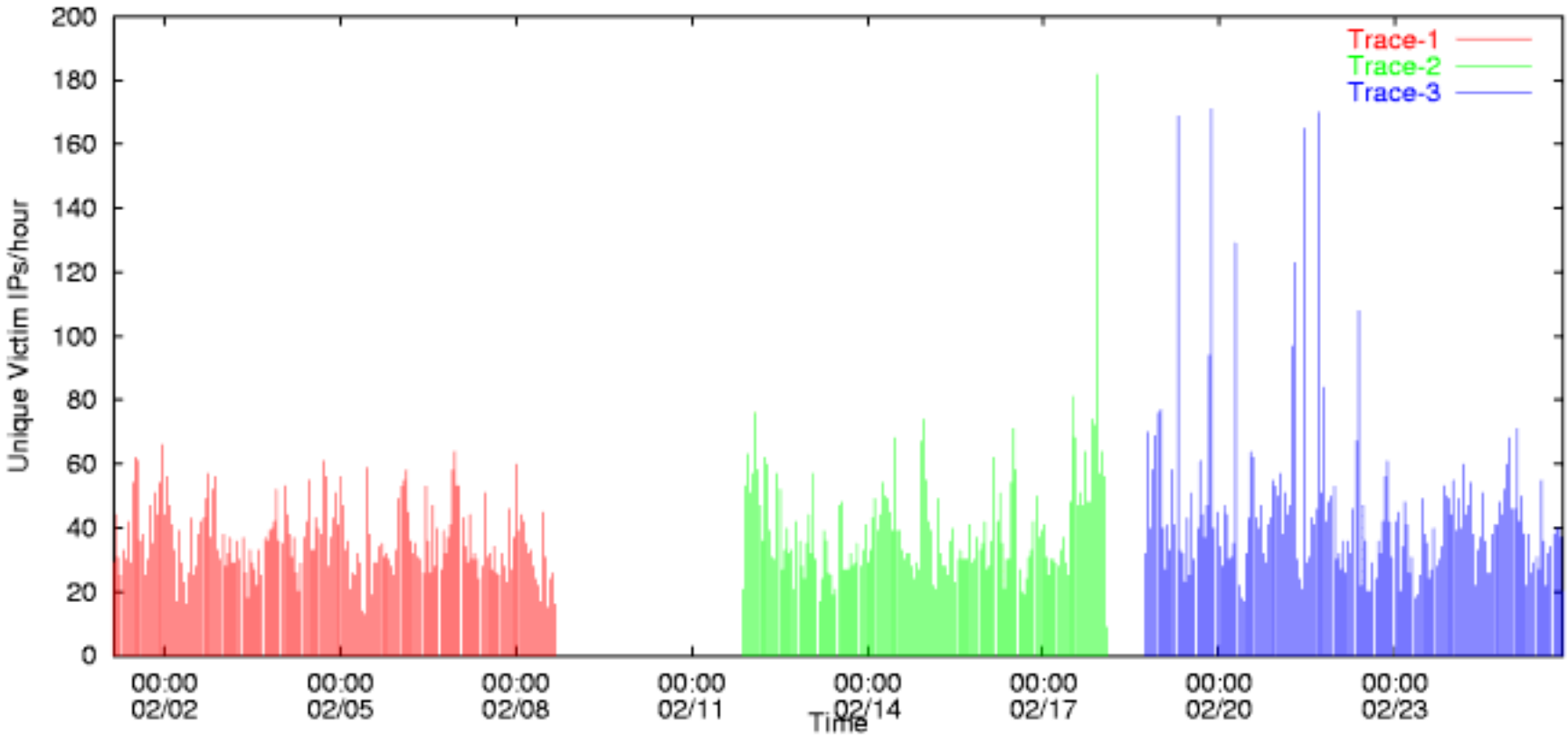
Example: random spoofing -> *backscatter*



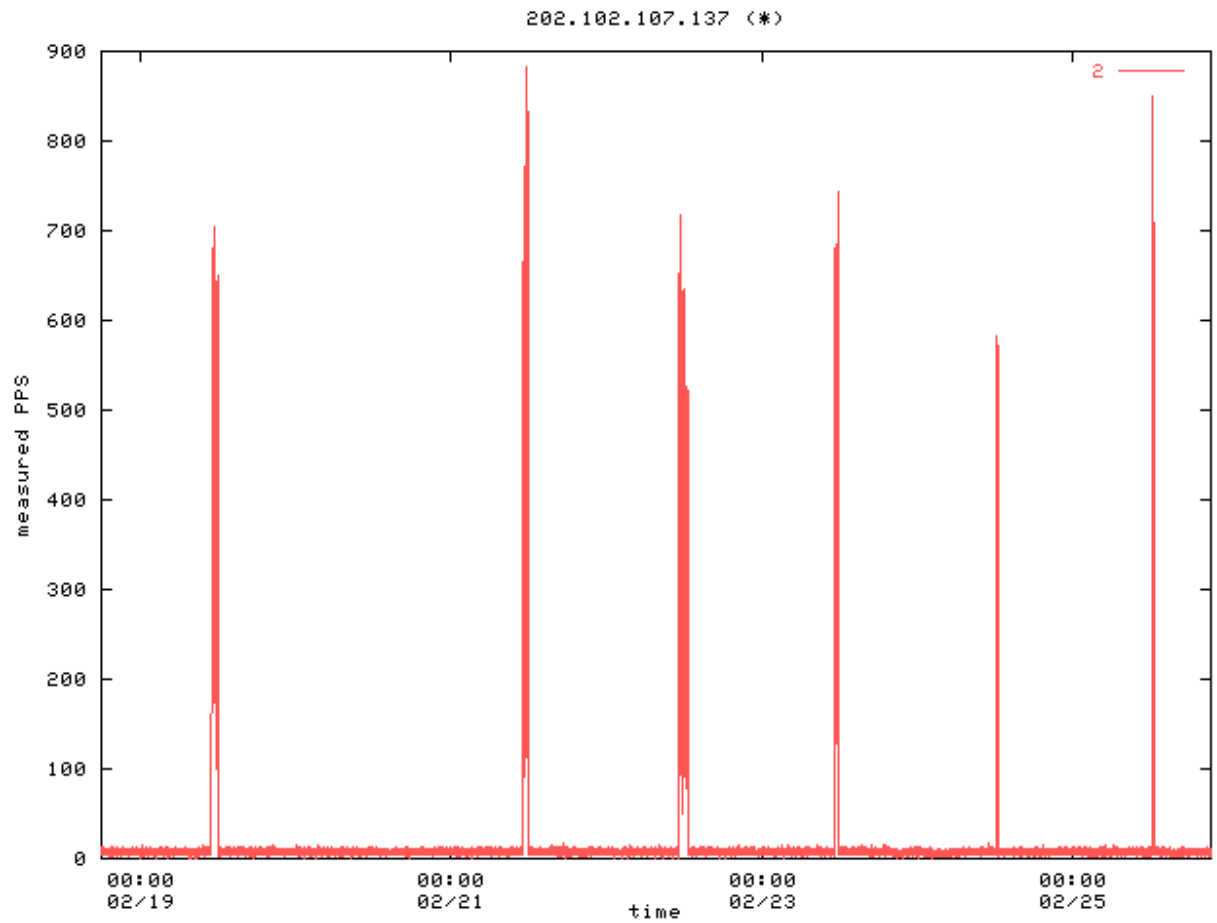
Our experimental apparatus…



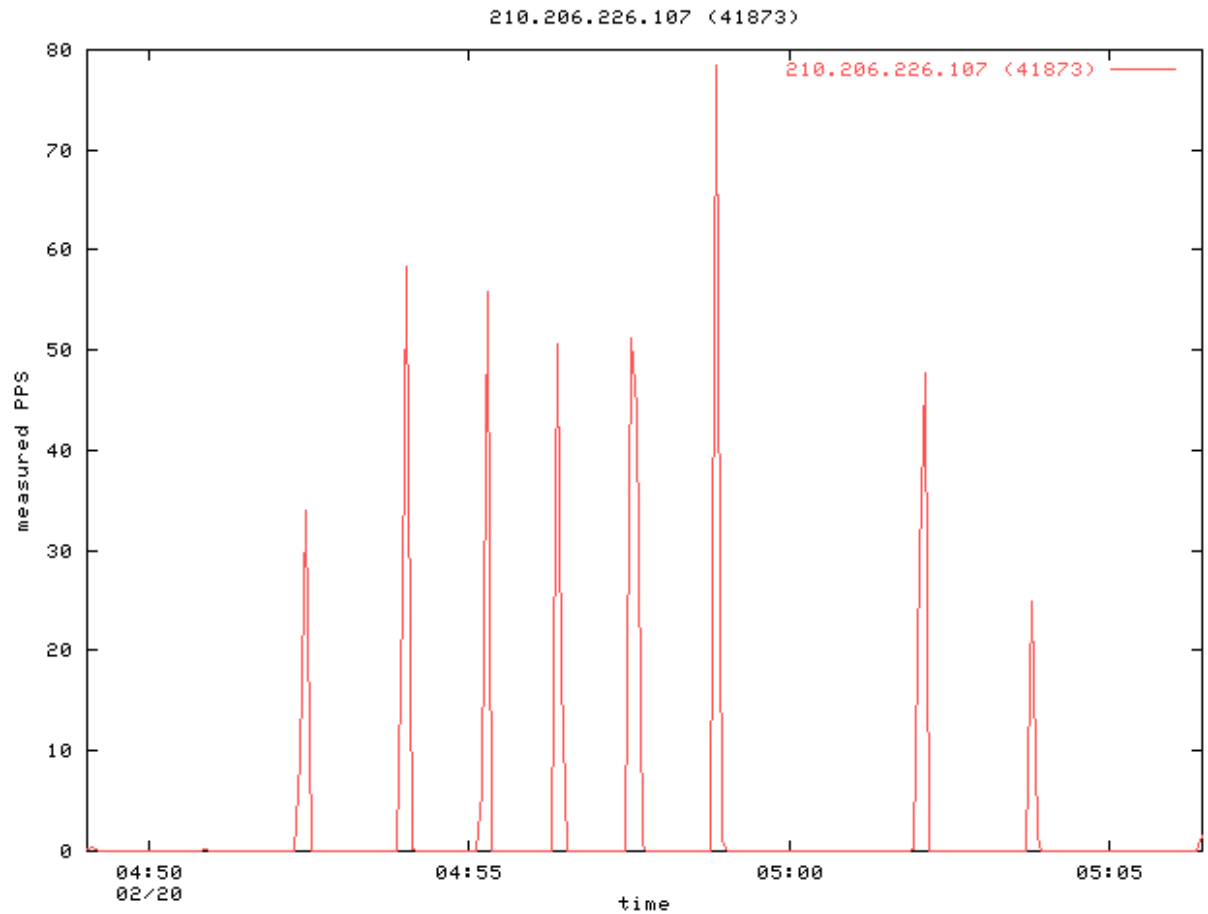
Attack volume over time



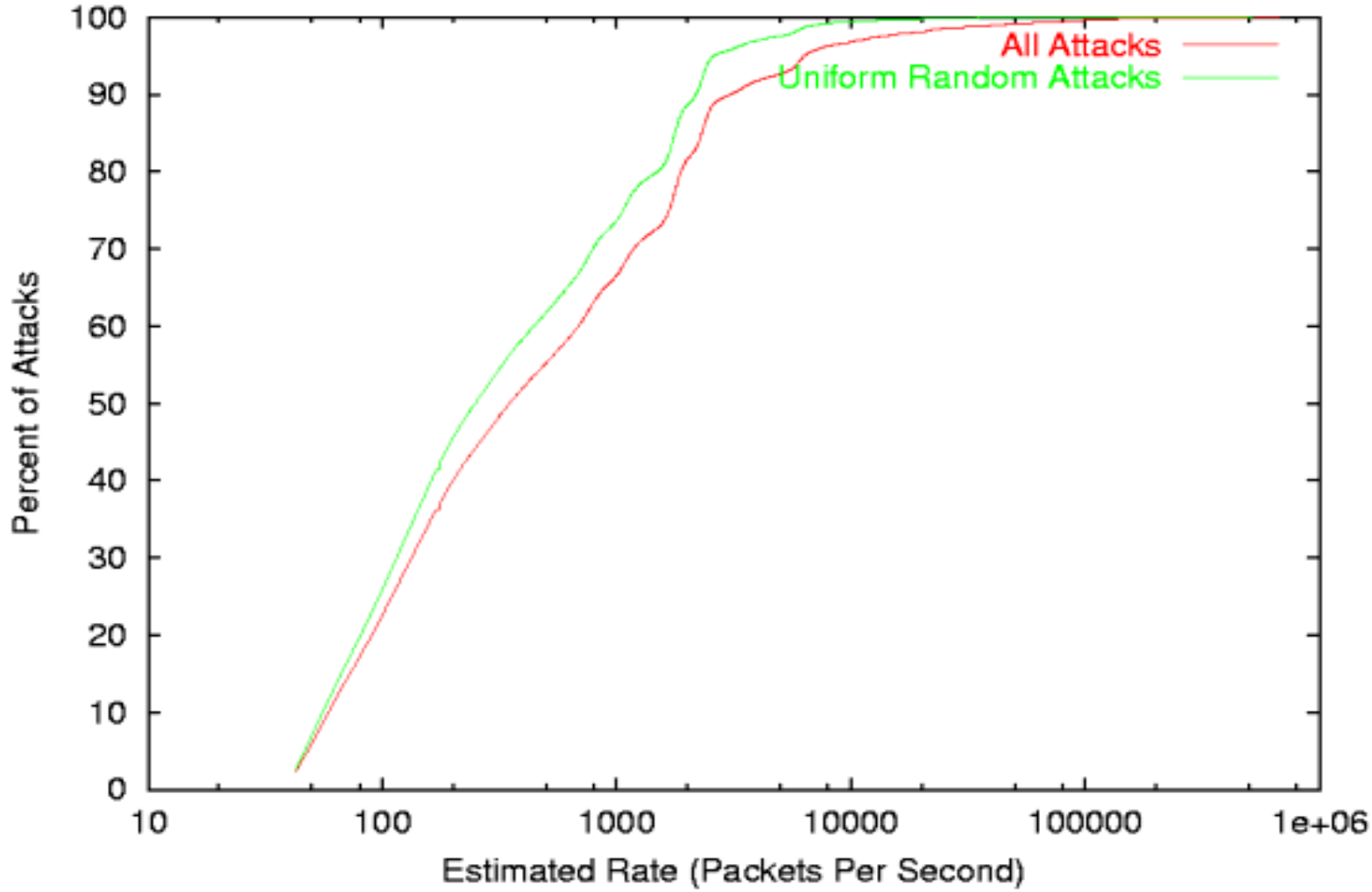
Example: Periodic attack (1hr per 24hrs)



Example: Punctuated attack (1min interval)



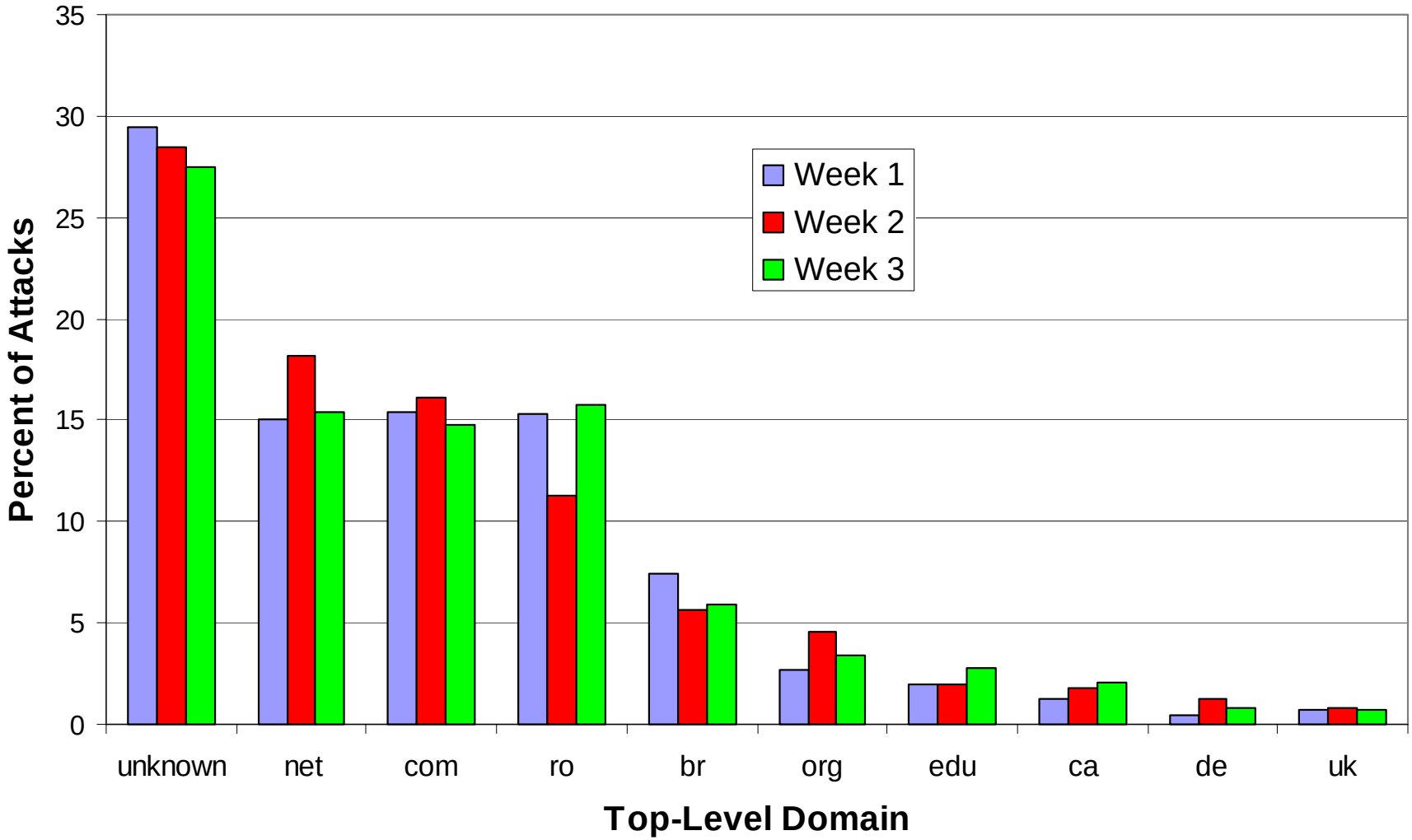
Attack rate distribution



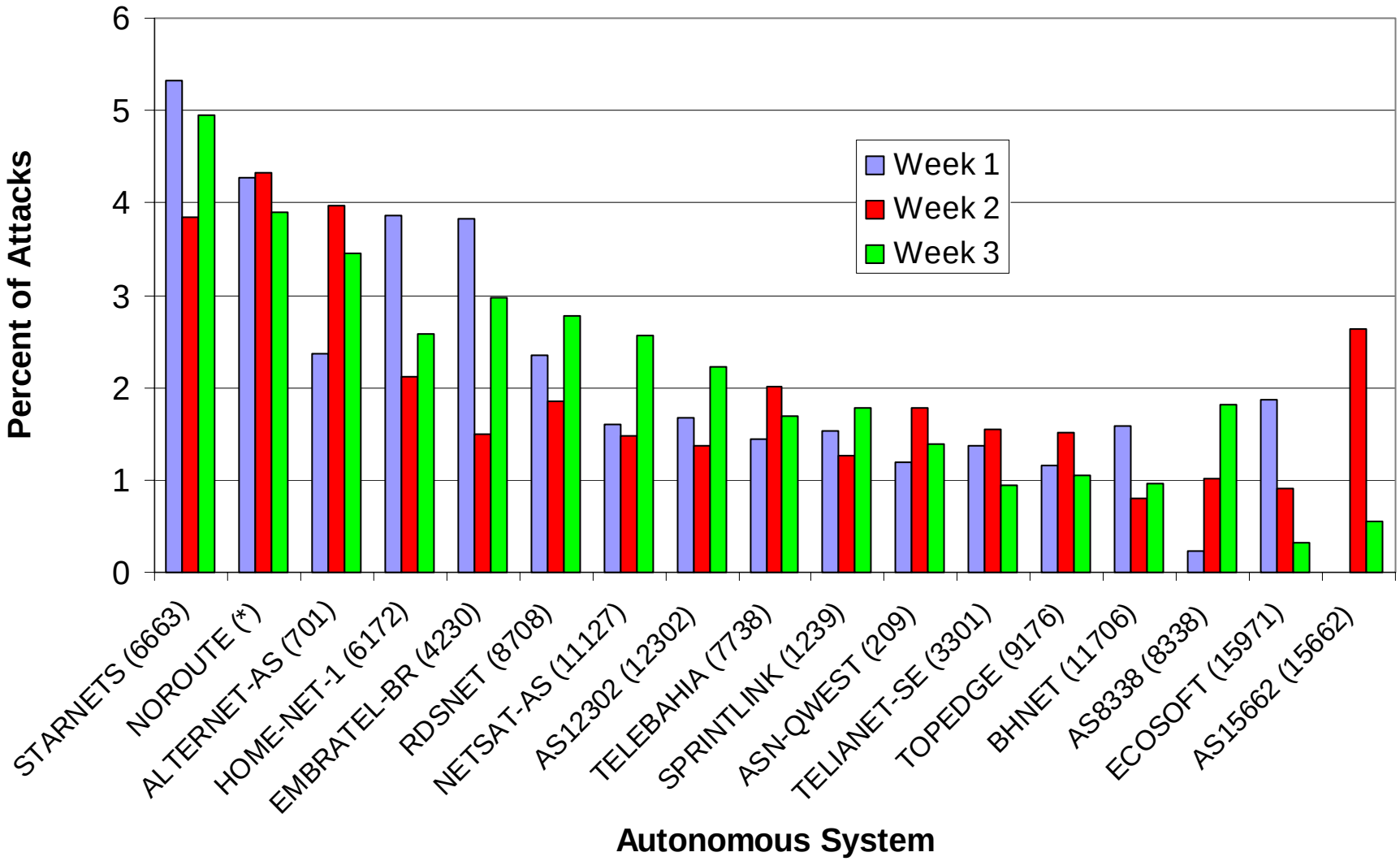
Victim characterization by DNS name

- Entire spectrum of commercial businesses
  - Yahoo, CNN, Amazon, etc. and many smaller businesses
- Worldwide phenomenon (>70 countries)
- Attacks on individuals
  - 10-20% of attacks on home machines
  - A few very large attacks against broadband
- 5% of attack target **infrastructure**
  - Routers (e.g. core2-core1-oc48.paol.above.net)
  - Name servers (e.g. ns4.reliablehosting.com)

Victim breakdown by TLD



Victim breakdown by AS



Summary of key results

- Lots of attacks – some very large
  - **>12,000** attacks against **>5,000** targets in 3 weeks
  - Most <**1000** pps, but some over **600,000** pps
  - Analysis is **conservative**; actual is clearly even higher
- **Everyone** is a potential target
  - Targets not dominated by any TLD, 2LD or AS
    - Targets include large e-commerce sites, mid-sized business, ISPs, government, universities and end-users
- New attack behavior
  - Punctuated/periodic attacks
  - Attacks against infrastructure and broadband targets

Recent advances in DoS attacks

- Minimal innovation in DoS *content*
  - TCP (SYN, ACK/mstream, RST, **randomization**)
  - ICMP (particularly via Smurf)
  - UDP (DNS)
  - Fake encapsulations (GRE, IPIP)
- Significant innovation in DoS *control*
  - Encrypted control channel
  - Oblivious control
  - Leveraging existing communications medium (i.e. IRC)
- Innovation in DoS *distribution*
  - Highly automated probe and exploit engines
  - **Worms**

The CodeRed Worm: We were very lucky

- **CodeRed**: DoS tool mated with a virulent worm
  - Uses .ida exploit to take over IIS Web servers
  - Replicates by targeting random addresses
  - At synchronized time all infected servers flood www1.whitehouse.gov
- Why it didn't take down the Internet
  - Great worm, **poor DoS tool** + lots of *advance warning*
  - Targeted static IP address
    - whitehouse.gov moved, Genuity blackholed old IP
  - TCP-based attack required successful connection to victim
- Why it could have
  - > 300,000 hosts taken over in a day (CRv2)
  - Potential "firepower" is staggering (multiple Tbps)

Attack trends for next year

- Punctuated attacks
  - Avoids static detection triggers
- Target selection
  - Infrastructure (routers, DNS, DHCP, etc)
- Reflector attacks
  - Increased power, anonymity, amplification
- Dynamically shifting sources and attack type
  - Evade static filters
- Targeted address spoofing
  - Less obvious, harder to track
- Worms + flexible DoS tools + IRC control

Today's situation

- Attacks are increasingly widespread
- Automated attack tools are becoming more sophisticated faster than defenses
- Barrier to entry is steadily decreasing
- **Responding is slow and expensive**
  - Little automation in use today
  - The available pool of good security and network personnel is shrinking

What can be done?

- Prevention
  - Global “best practices” to make it harder for attacks to infiltrate and hide on our systems
  - http://www.sans.org/ddos_roadmap.htm

- Response
  - Forensic: catch the bad guy
    - Associate individual with attack and amass sufficient evidence to prosecute; *difficult and time-consuming*
  - **Operational**: stop the pain
    - Stop, block or counter attack; allow normal service to operate unimpeded

DDoS attack response phases

- **Detect**
  - Figure out you're being attacked and how
- **Locate**
  - Figure out where/how attack enters your network
- **Counter**
  - Keep attack packets from reaching victim

Detection

- Key problem
  - Differentiating attack from a lot of legitimate traffic
- State of practice
  - Manual examination of traffic monitors + packet sniffer output
  - IDS signatures on zombie/handler communication (limited)
- State of art
  - **Signature-based** traffic characterization
    - Few false positives, lots of false negatives
  - **Anomaly-based** traffic characterization
    - Packet “type” distributions
    - Protocol dynamics and “rules”
    - Multi-site correlation
    - Short-term and long term traffic trends

Location

- Key problem
  - Which routers and links does the attack traverse?
- State of practice
  - Manual, hop-by-hop inspection of router logs (e.g. IOS "*log input*")
- State of art
  - Automatic traceback using statistical data (e.g. Netflow) and multi-device correlation
    - Use attack characterization + topology to check which links forwarded suspect traffic to victim
  - Special case: attacks with *random source addresses*
    - Determine ingress by blackholing target and internally routing unallocated "canary" prefixes

Countermeasures

- Key problem
  - How to block or counter attack?
- Disrupt source
  - Exploit zombie flaws or imitate handler (e.g. ZombieZapper)
  - Not a long-term solution
- Restrict attack
  - Blackhole (remove route) for target IP
    - Sacrifice host to save link
  - Classify and filter attack (ACLs and rate-limiters)
    - Finer grained control, but more overhead
  - Re-route

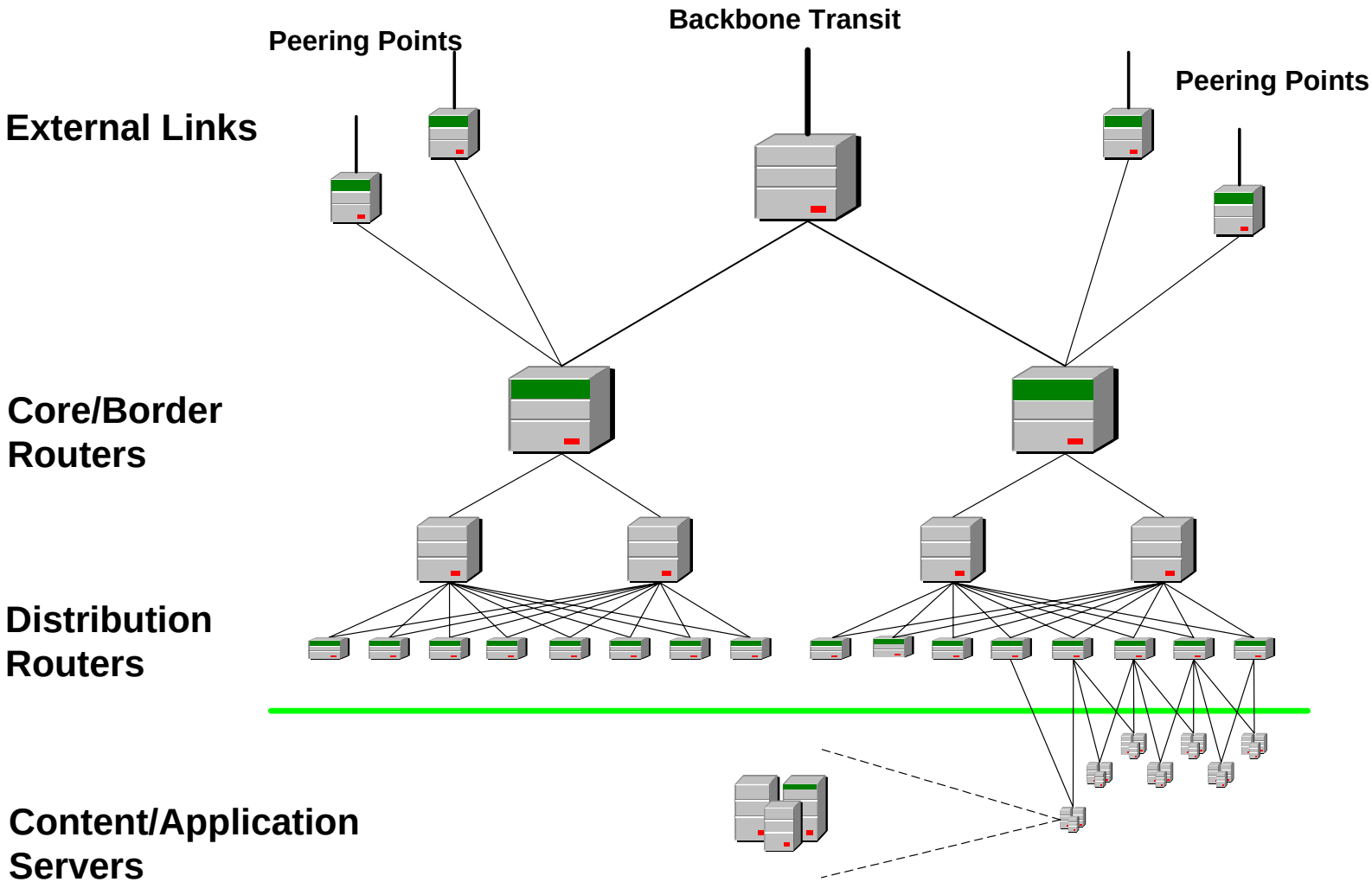
Filtering DoS traffic

- Construct filters to maximally block attack and minimally impact good traffic
  - Goal: best match filter, on router(s)/switch(es) closest to attack ingress, with lowest forwarding impact
- Optimization issues
  - Constructing “best match” filter
  - Where to place filter in topology
  - Overhead of executing filter on interface

Optimizing for equipment capabilities

- Overhead
  - Classifier performance vs. complexity vs. line rate
  - Distributed vs. centralized implementation
- Limited syntax
  - Some boxes can classify packets on arbitrary fields and integer ranges, others have limitations
- Filter actions
  - Packet dropping
  - Shaping vs. rate-limiting vs. pure priority
  - **Provisioning** is special case of rate shaping
  - Rerouting

Optimizing for topology: where to filter?



The missing links… tying it together

- Automated monitoring and analysis
  - Monitor data across entire network
  - Automatically detect, locate and solve countermeasure optimization problems
- *Aided* human oversight
  - Human-sensible evidence and policy control
  - Explicit manual control of recommended countermeasures
- Scalability
  - Handle large line rates (GE, OC48 and above)
  - Support large networks (1000's of elements)
- Customer/provider communication
- OSS integration

Conclusion

- Denial-of-Service is a tough problem
- There are a lot of attacks at any given time
- Attacks are increasing in magnitude and sophistication
- The key to defense is *knowledge* and *speed*
  - Automated attack detection, diagnosis, location
  - Semi-automated countermeasures

UCSD Study: Assumptions and biases

- *Address uniformity*
  - Ingress filtering, reflectors, etc. cause us to **underestimate** # of attacks
  - Can bias rate estimation (can we test uniformity?)
- *Reliable delivery*
  - Packet losses, server overload, and rate limiting cause us to **underestimate** attack rates/durations
- *Backscatter hypothesis*
  - Can be biased by purposeful unsolicited packets
    - Port scanning (minor factor at worst in practice)
  - Do we detect backscatter at multiple sites?

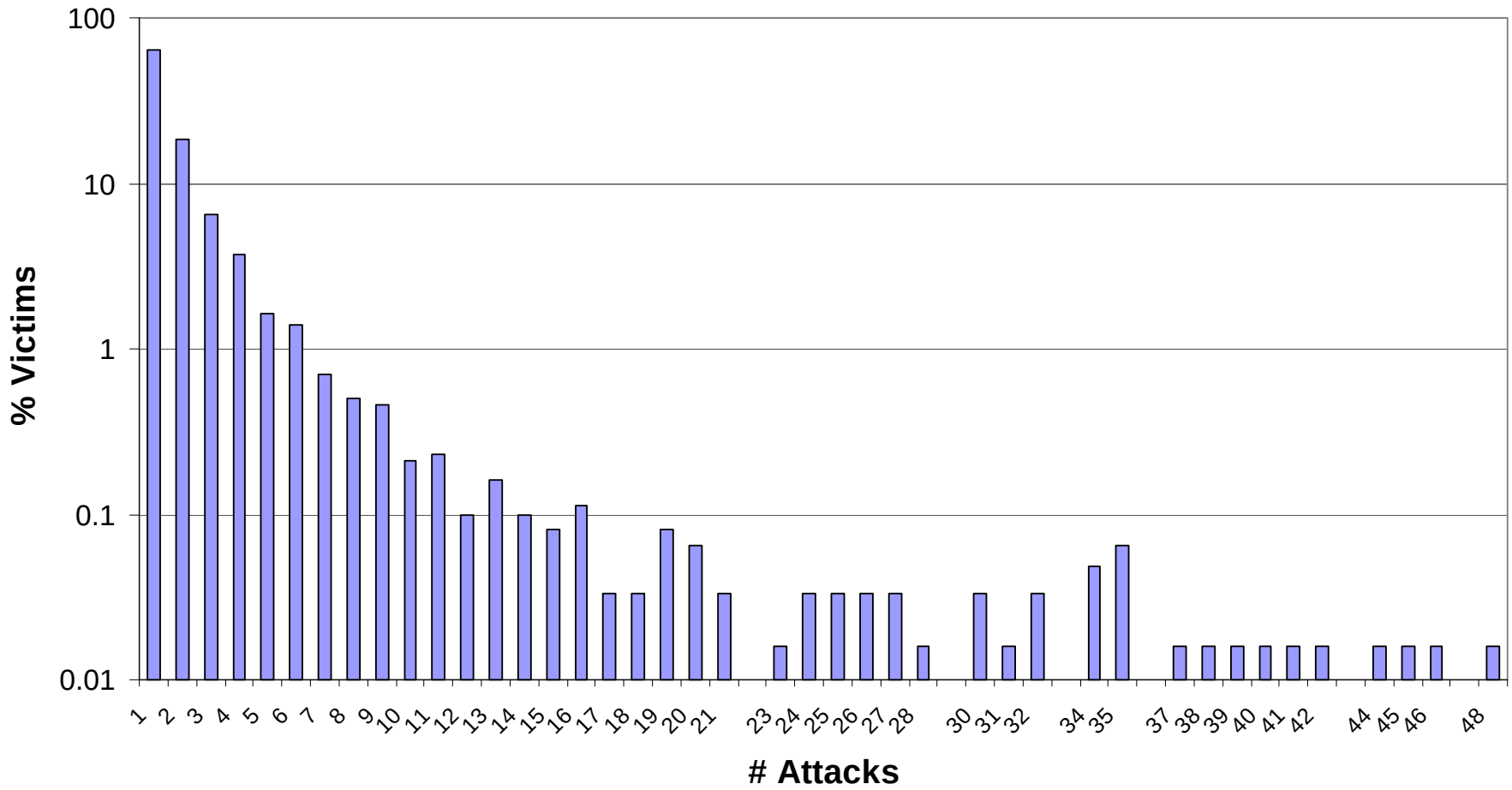
UCSD Study: Validation

- Backscatter not explained by port scanning
  - 98% of backscatter packets don't cause response
- Repeated experiment with independent monitor (3 /16's from Vern Paxson)
  - Only captured TCP SYN/ACK backscatter
  - 98% inclusion into larger dataset
- Matched to actual attacks detected by Asta Networks on large backbone network

Identifying attacks

- Flow-based analysis (categorical)
  - Keyed on victim IP address and protocol
  - Flow duration defined by explicit parameters (min threshold, timeout)
- Event-based analysis (intensity)
  - Attack event: backscatter packets from IP address in 1 minute window
  - No notion of attack duration or “kind”

Distribution of repeat attacks



Backscatter protocol breakdown (one week)

Backscatter protocol	Attacks	BS Packets (x1000)
TCP (RST ACK)	2027 (49)	12,656 (25)
ICMP (Host Unreachable)	699 (17)	2892 (5.7)
ICMP (TTL Exceeded)	453 (11)	31468 (62)
ICMP (Other)	486 (12)	580 (1.1)
TCP (SYN ACK)	378 (9.1)	919 (1.8)
TCP (RST)	128 (3.1)	2,309 (4.5)
TCP (Other)	2 (0.05)	3 (0.01)

Attack protocol breakdown (one week)

Attack Protocol	Attacks	BS Packets (x1000)
TCP	3902 (**94**)	28705 (56)
UDP	99 (2.4)	66 (0.13)
ICMP	88 (2.1)	22,020 (**43**)
Proto 0	65 (1.6)	25 (0.05)
Other	19 (0.46)	12 (0.02)