Distributed Denial of Service Attacks

Grant Goodale
Cornell University
November 2004

A Brief Introduction

- What is a DoS Attack?
  - A explicit attempt by attackers to prevent the legitimate use of a service.
    - Targets include both end hosts and infrastructure
  - Distributed DoS: use of multiple machines to execute a DoS attack
  - Long history (~1996), many techniques

* http://www.cert.org/tech tips/denial of service.html

A Short History of DDoS

- Up to 1996: point to point
    - SYN flooding, PoD, fragmentation attacks
- 1997-1998: combined attacks
    - smurf, fraggle, teardrop, winnuke
    - Increasing sophistication in deployment, ease of use

A Short History of DDoS

- 1999-2000: Flooding, IRC
  - ip-proto-255, TCP NULL flood
  - Encrypted custom C&C channels, IRC
  - payload includes remote shell, auto-update
- rootkits start including DDoS toolkits

A Short History of DDoS

- 2001-2002: Reflection attacks, worms, Countermeasures
  - Code Red, l10n
  - Scan, infect, repeat
  - IRC channel hopping

A Short History of DDoS

- 2003-2004: Blended threats, sophisticated delivery
  - Windows vulnerabilities (RPC DCOM, etc.) provide easy attack vector for worms (Slammer)
  - More valid traffic (non-spoofed source IP, randomized valid payload
- Hard to distinguish between attack and ‘Flash Crowd’

A Growing Problem

- Estimates of “hundreds of attacks a day” - in 2001
- New trend: networks of machines for hire
  - Send spam during the day, attack your competitors at night
- Toolkits require almost zero skill to use - just download and start 0wning machines

The DDoS Arms Race

- On one side: Solution Providers, ISPS, Academia
- On the other:

Defense Techniques

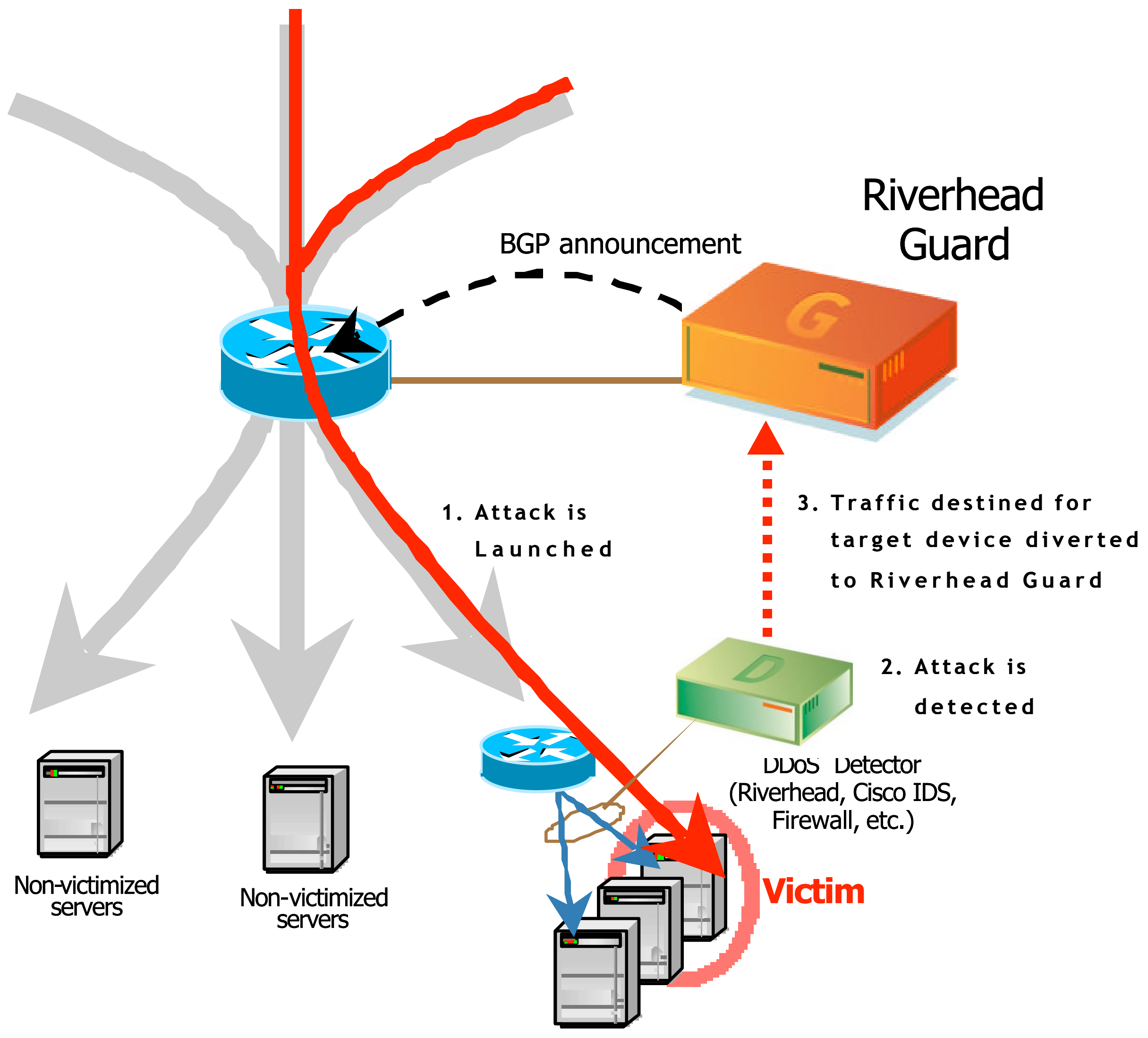
- Axes of comparison:
  - Passive (detection) vs. active (prevention)
  - Edge-deployed vs. target-deployed
- Common approaches:
  - Packet filters at the edge
  - Traffic characterization

The Papers

- Riverhead Networks' Traffic classification and filtering system
- Riverhead Networks' Long Diversion method of centralized DDoS protection
- Firebreak - routing/trusted intermediary solution
- Why these papers? Practical, not theoretic solutions

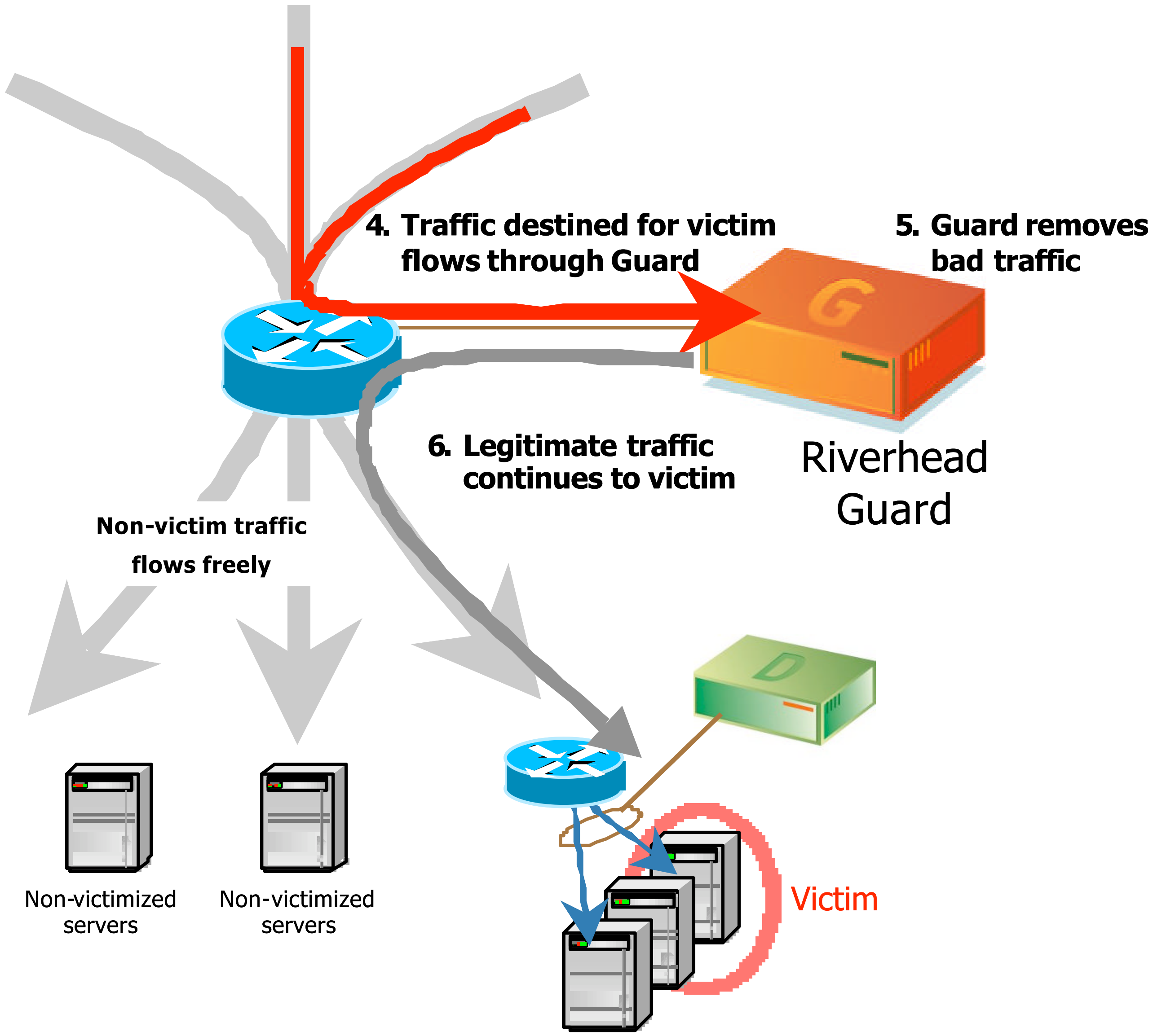
Riverhead Networks

DDoS attack results in BGP announcement diverting traffic away from the target to the Riverhead Guard

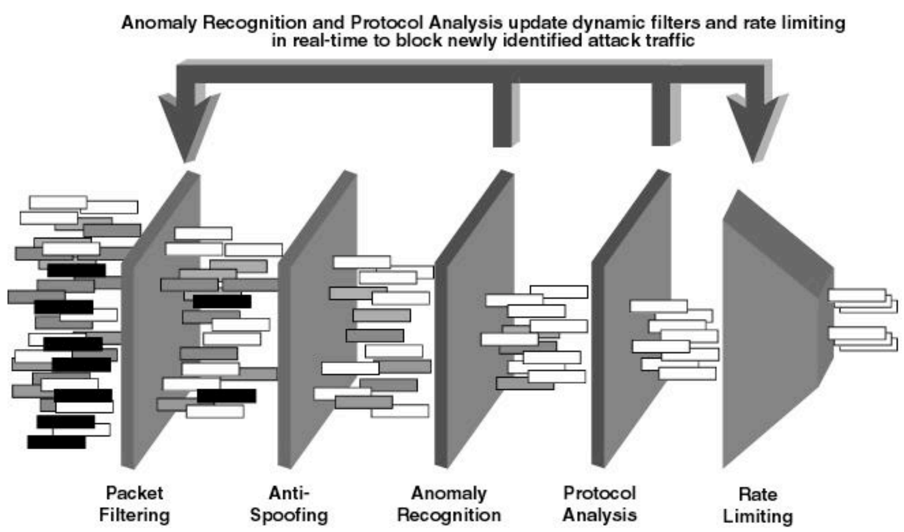


Riverhead Networks

Traffic is filtered based on several criteria; "good" traffic then sent back to target

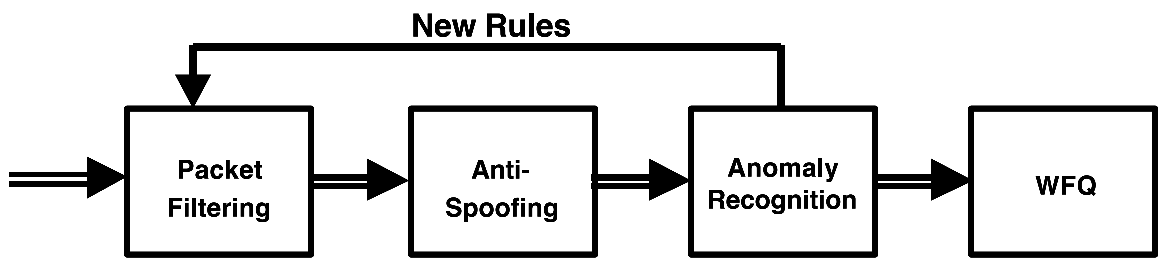


Riverhead Networks



- Five-step pipeline
- Heuristic analysis based on traffic modeling
- WFQ rate limiting as last step

Riverhead Networks



- Heuristic analysis applied traffic after spoofed packets are removed
- Results are fed back into the initial packet filters
- Analysis at network and application layer

Difficulties

- Effective filtering requires non-noisy training data
  - How ‘human’ can attack traffic be made to look?
- Deployment does not reduce traffic on network upstream from target
- Scalability issues - not clear what server/Guard ratio is necessary

Riverhead Long Diversion

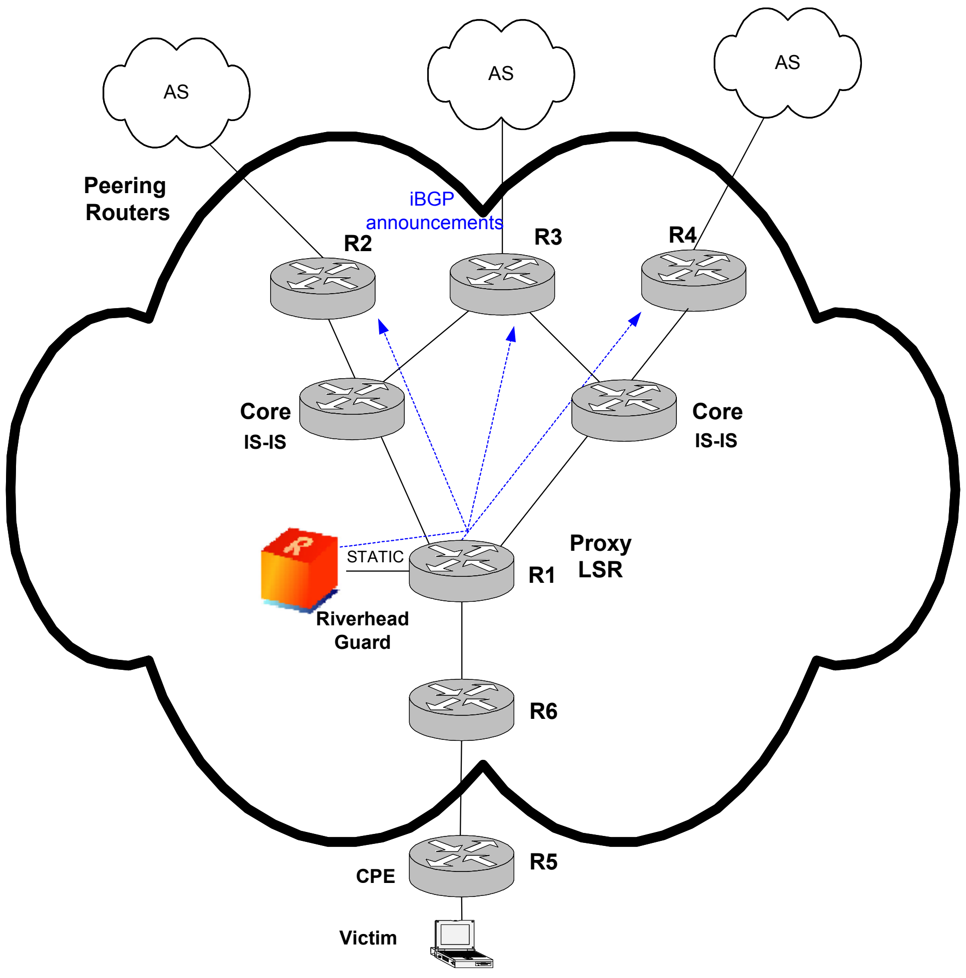
$$
\begin{array}{r|l}
 & 25 \\
\hline
17 & 425 \\
 & \underline{-34} \\
 & 85 \\
 & \underline{-85} \\
 & 0
\end{array}
$$

Riverhead Long Diversion

- Use MPLS to create a LSP from peering point to Guard when attack is detected
- One Guard can interface with several peering points
- Billed as a cost-effective solution for ISPs to sell to SMB market

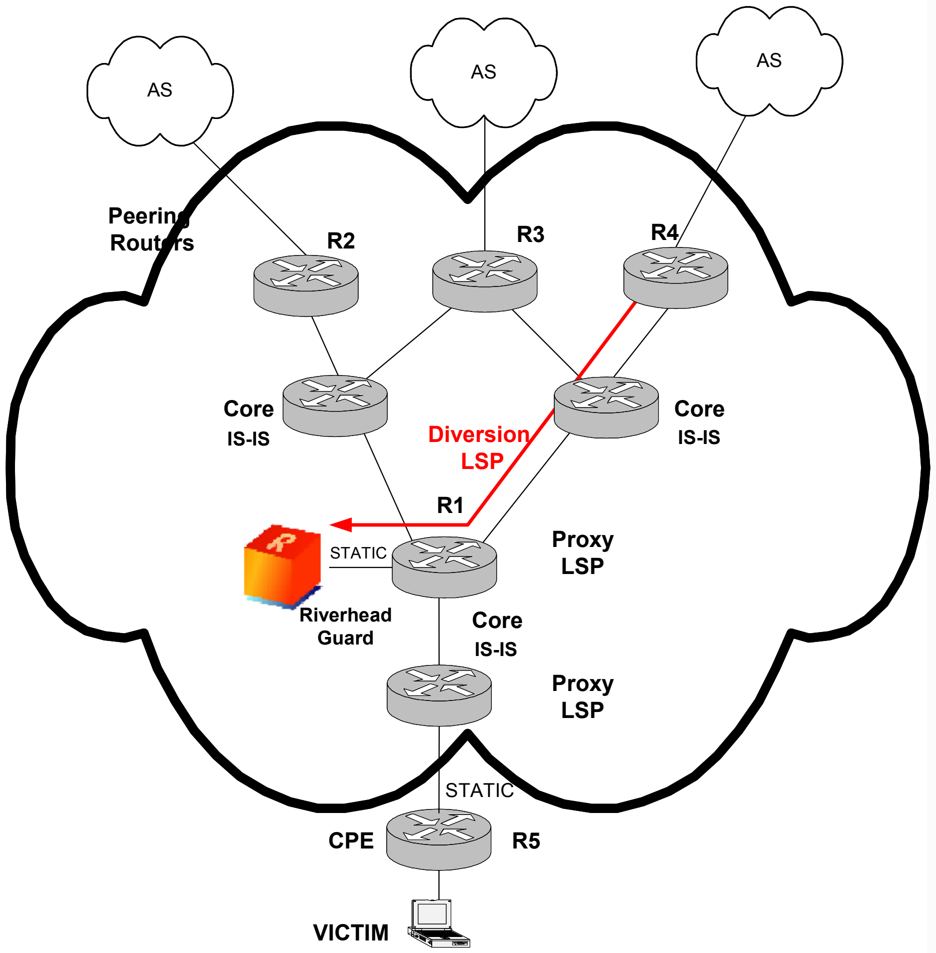
Riverhead Long Diversion

- Riverhead Guard sends iBGP announcement to peering routers
- prefixes are longer than previously advertised prefix for target



Riverhead Long Diversion

- Traffic for target host is diverted to to Guard via newly created LSP
- Guard performs traffic analysis and forwards valid traffic on to original destination



Difficulties

- Again, upstream provider(s) still have to carry attack traffic
- Now have to maintain *n* traffic models (or come up with a global representation for ‘normal’ traffic
- Customers’ DDoS protection is now interrelated (More complex SLA and assurance necessary)

Firebreak

- Fabulous!
- Wonderful.
- Have a good night. Drive safe.
- Look, a bird!
- But seriously...

Firebreak

- Targets protected by group of boxes deployed near the edge - firebreaks
  - ISPs deploy firebreaks in POPs - as close to the customer as possible
- Target's IP address is not routable except from firebreak machines
- How do clients connect?

Firebreak

- Firebreak hosts map anycast firebreak addresses for each protected target to reachable target addresses (IP-level indirection)
- Only packets from firebreak machines are routable to protected targets
- Firebreaks are themselves firebreak-protected

Firebreak

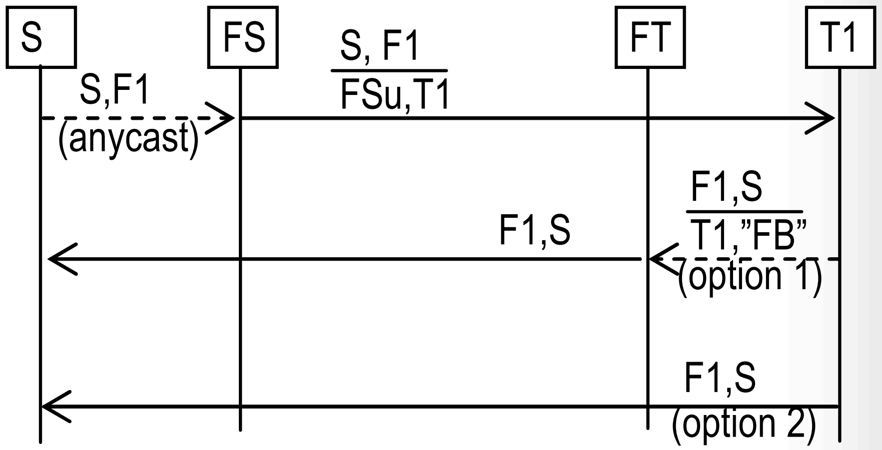
- Target is also protected by DDoS detector
  - Feedback is provided to firebreak nodes in the event of attack (handling is TBD and probably application-specific)
- Target's outbound traffic is a problem - how does the target communicate with other hosts (both protected and unprotected)?

Firebreak

- Two possibilities:
- Spoof source IP with firebreak anycast address
  - Could cause problems if edge routers are configured to drop spoofed packets
- Tunnel all outbound traffic through a nearby firebreak
  - potential bottleneck / single point of failure

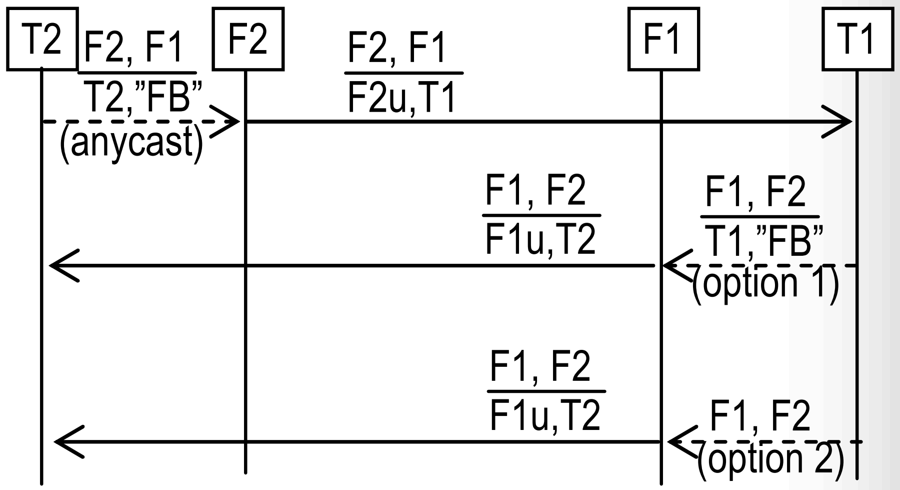
Firebreak

- Client (S) addresses packets to nearby firebreak (FS) using anycast address for server T1
- FS maps anycast address to privately address for T1.
- T1 responds using anycast address as source, or routes response through firebreak



Firebreak

- How do two protected hosts communicate? Via each other's firebreak addresses
- Neither side needs to know whether the other is protected
- Requires firebreak handling of all traffic between protected hosts



A Few Issues

- All minor. Really!
- Two addresses per target (not a problem with IPv6)
- Anycast isn’t widely deployed - significant infrastructure to design/test/implement
  - (and/or convince Akamai)
- Scaling issues (esp. with outbound traffic tunneling)

A Few Issues

- High degree of complexity
  - Many moving parts over WAN
  - Interactions between multiple firebreaks, targets, DDoS detectors, &c.
  - Do ASes interact with each other's firebreak systems?
- Isolation of DDoS sources still difficult

Compare and Contrast

- Both solutions use routing to protect the target from attack
- Riverhead approach still makes server IP public
  - Does not require extra processing of outbound traffic
  - Non-flooding attacks may not be detected
  - Firebreak can protect from all IP attacks

Compare and Contrast

- Riverhead attempts to put prevention logic close to the destination; Firebreak pushes it out (near) to the source
    - Firebreak is better for overall network utilization
- Both solutions have separate detection and control components

Other Solutions: Akamai

- Origin server IP address kept secret
  - Security through obscurity!
- Two tiers of DNS
  - Dozens (?) of top tier servers, reached by IP anycast. Large TTL.
  - Thousands (?) of second tier servers. Small TTL.
  - This is "quite good" protection *

* http://www.cs.cornell.edu/People/francis/firebreak/firebreak-june-04-v2.pdf. All conclusions theoretic.

Other Solutions: Akamai

- Potential problems:
    - Attack origin servers by discovering IP address(es)
        - "Static" content cached at Akamai proxies ok
        - Akamai could reconfigure those addresses...

Other Solutions: Akamai

- Potential Problems (cont.):
  - Sustained attack on top tier of DNS
    - But ISPs can traceback attackers and install filters on timescale of hours
    - But if this attack succeeds, all Akamai customers are denied service!

Questions?

Sources

- http://sfs.poly.edu/presentations/Crash_Course_in_DDoS.ppt
- http://staff.washington.edu/dittrich/l2-ddos.ppt
- http://www.cs.cornell.edu/people/francis/firebreak/firebreak-june-04-v2.pdf