The Case for IP Puzzles

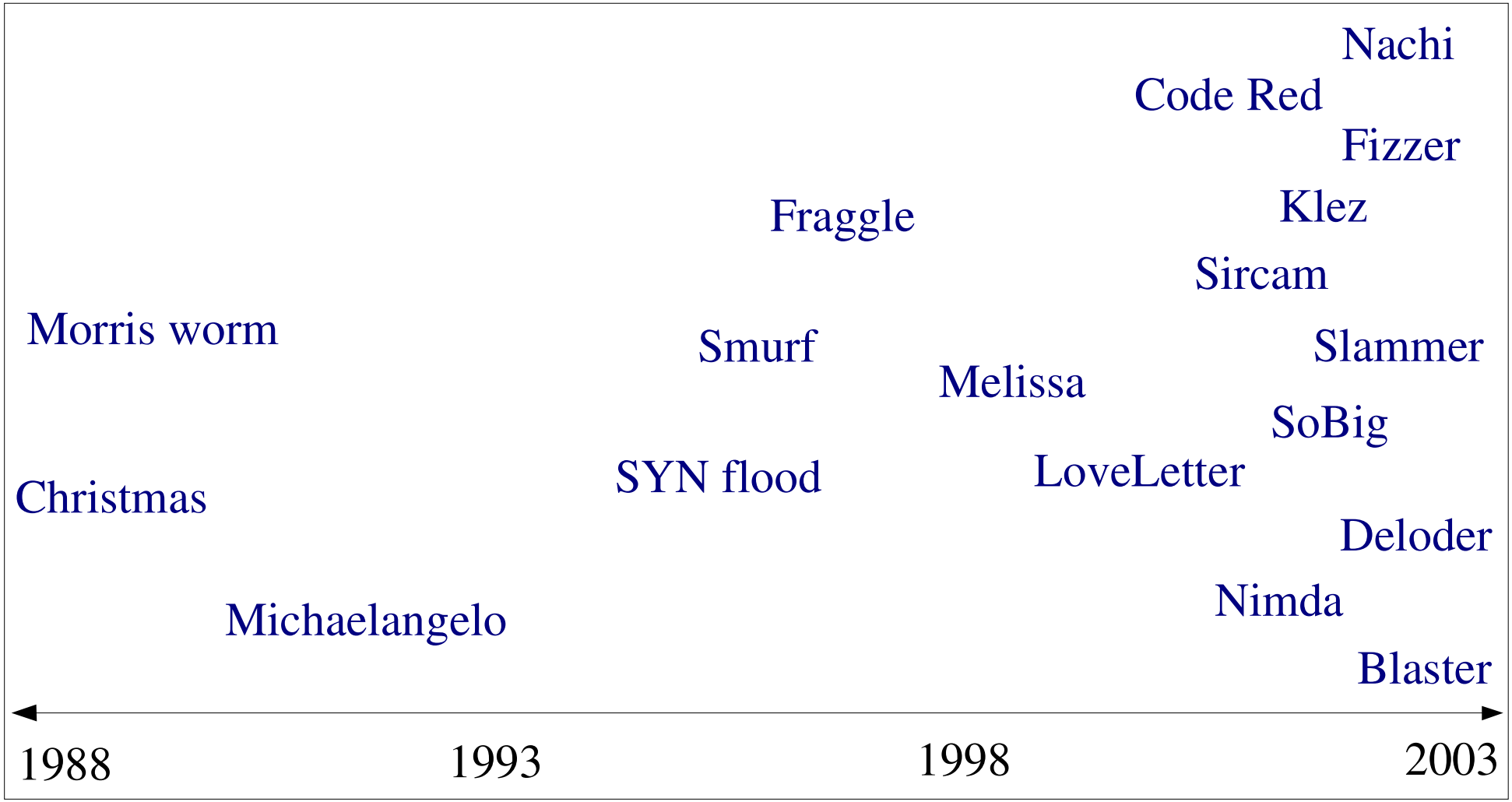
Wu-chang Feng

OGI SCHOOL OF SCIENCE & ENGINEERING
OREGON HEALTH & SCIENCE UNIVERSITY

Motivation

- A quick look back on 15 years of not so “Good Times”

  SMTP, TCP, ICMP, UDP, FastTrack, SMB, finger, SSL, SQL, etc.



Puzzles

- An interesting approach for mitigating DoS activity...
  - Force client to solve a problem before giving service
  - Currently for e-mail, authentication protocols, transport layers
  - Fundamentally changes the Internet's service paradigm
    - Clients no longer have a free lunch
    - Clients have a system performance incentive to behave
- A contrast in approaches
  - Leave doors open and unlocked, rely on police/ISPs
    - Centralized enforcement (not working)
  - Give everyone guns to shoot each other with
    - Distributed enforcement (may not work either)
    - Harness the infinite energy of the global community to fight problem
    - Promising anecdotal evidence with spamming the spammers...

Posit

- Puzzles can only be effective if placed at the IP layer

Why are IP puzzles a good idea?

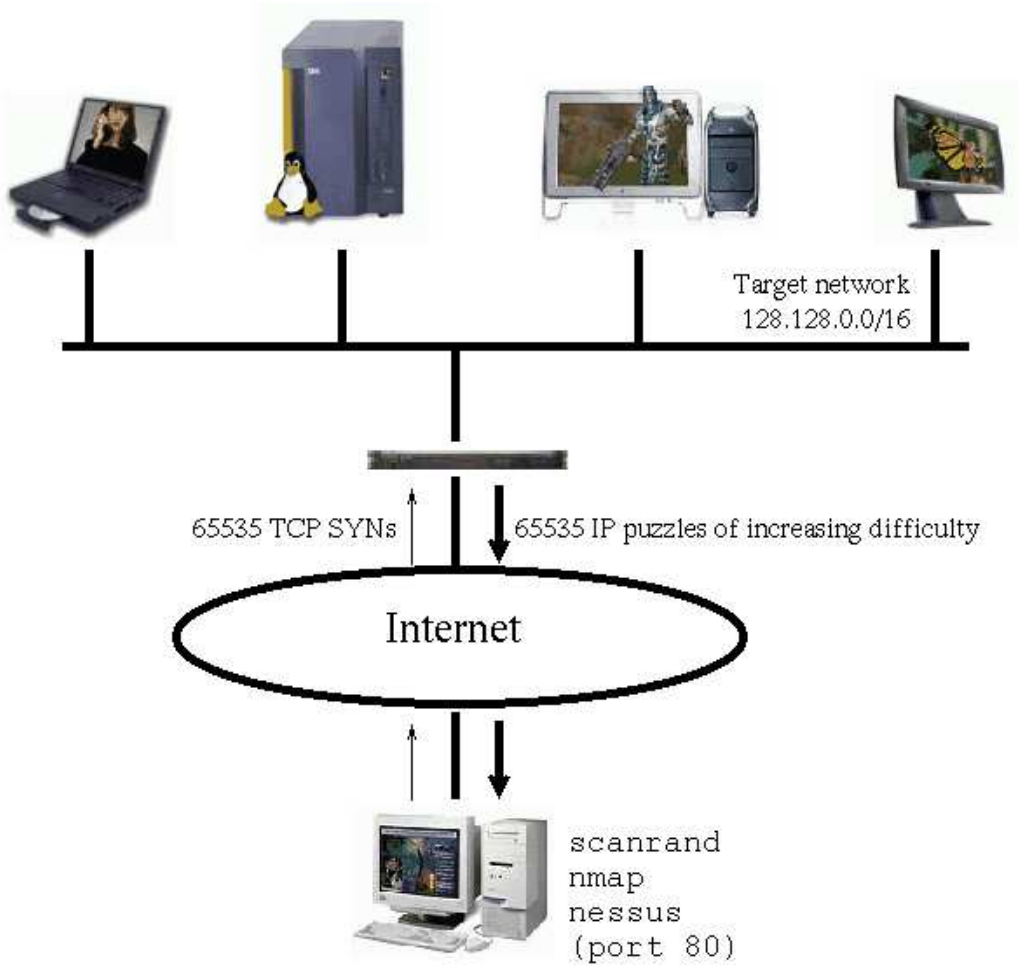
- "Weakest link" corollary to e2e/waistline principles
  - Put in the common waistline layer functions whose properties are otherwise destroyed unless implemented universally across a higher and/or lower layer
  - DoS prevention, congestion control destroyed if any adjacent or underlying layer does not implement it
    - TCP congestion control thwarted by UDP flooding
    - DoS-resistant authentication protocols thwarted by IP flooding
  - Until puzzles are in IP, it will remain one of the weakest links

IP puzzle scenario #1

- Port and machine scanning
  - Instrumental to hackers and worms for discovering vulnerable systems
  - The nuclear weapon: `scanrand`
    - Inverse SYN cookies and a single socket
    - Statelessly scan large networks in seconds
      - 8300 web servers discovered within a class B in 4 seconds
    - Technique not used in any worm....yet
      - Forget Warhol
      - “American Pie” worm => done in 15 seconds?
      - A grand networking challenge!

IP puzzle scenario #1

- Mitigation via a “push-back” puzzle firewall

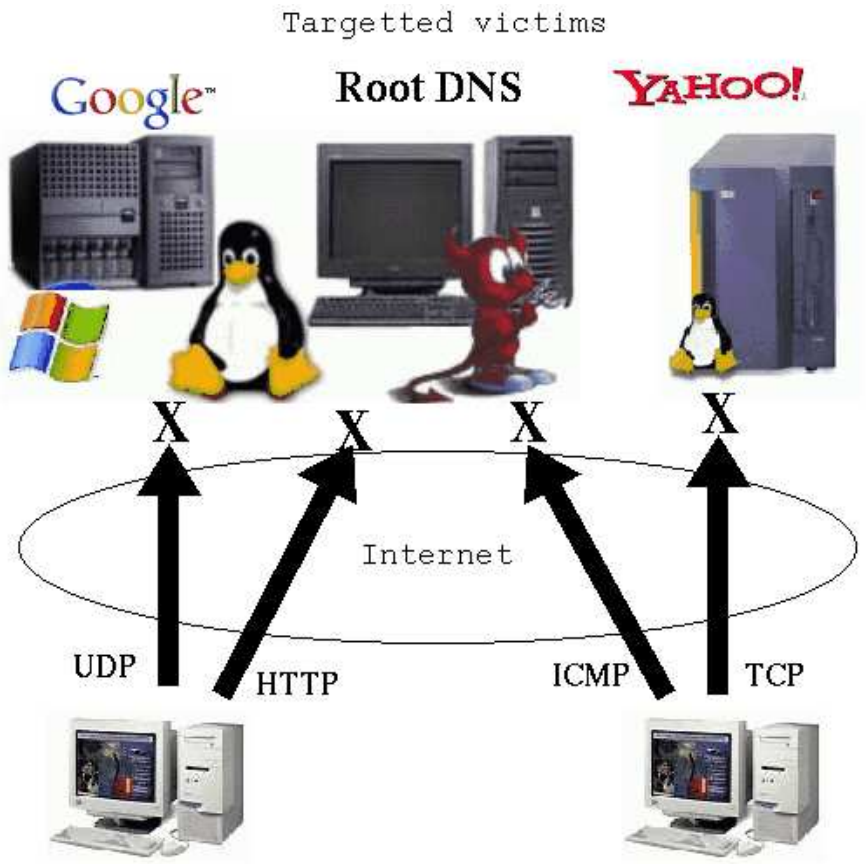


IP puzzle scenario #2

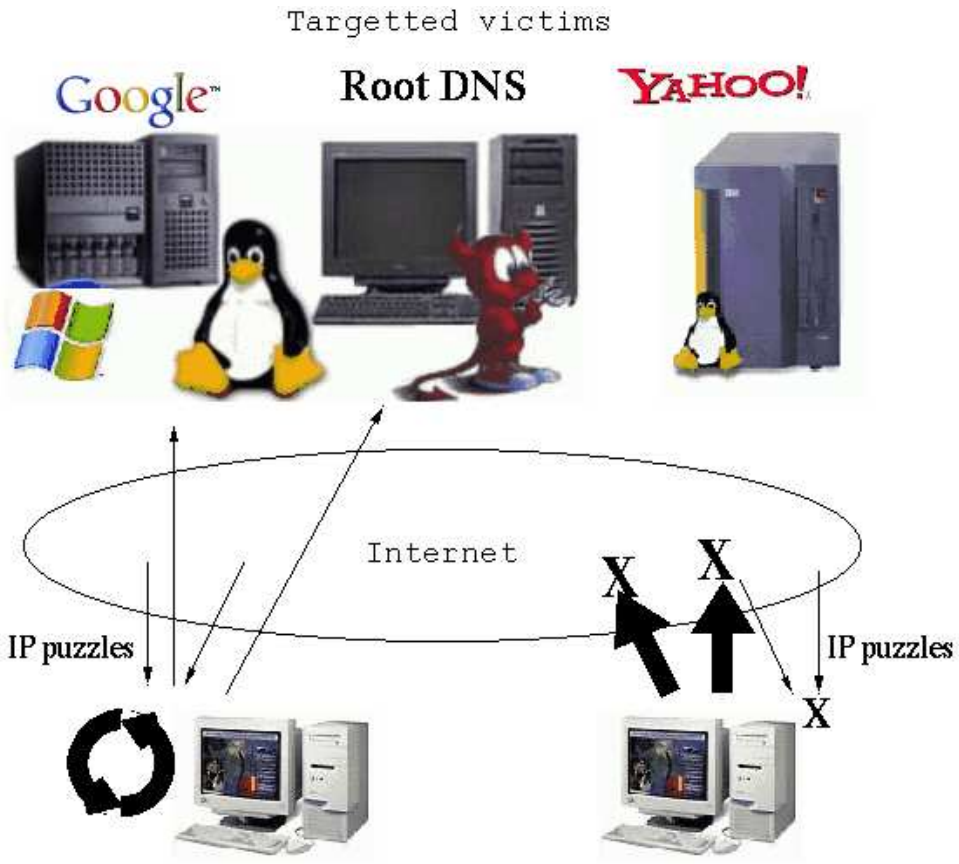
- Coordinated DDoS: simultaneous attacks against multiple sites from the same set of zombie machines
  - Mafiaboy (2000)
  - Have zombies initiate low bandwidth attacks on a diverse set of victims to evade localized detection techniques (such as `mod_dosevasive`)

IP puzzle scenario #2

- Mitigation using IP puzzles



Zombie participants in a coordinated DoS attack



Zombie participants in a coordinated DoS attack

Why are IP puzzles a bad idea? (What are the research challenges?)

- Tamper-resistance
- Performance
- Control
- Fairness
- Deployment

Tamper-resistance

- A tool to both prevent and initiate DoS attacks
  - Disable a client by...
    - Spoofing bogus puzzle questions to it
    - Spoofing its traffic to unfairly trigger puzzles against it
  - Disable a router or server by...
    - Forcing it to issue loads of puzzles
    - Forcing it to verify loads of bogus puzzle answers
    - Replaying puzzle answers at high-speed
  - Probably many more....

Performance

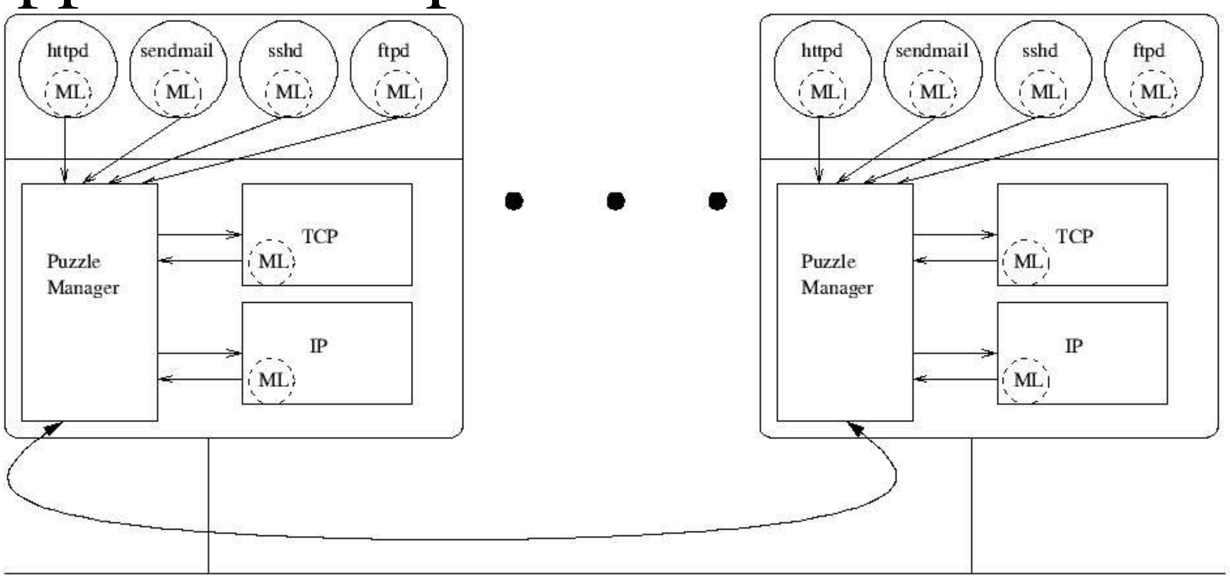
- Must support low-latency, high-throughput operation
  - Must not add latency for applications such as on-line games
  - Must support high-speed transfers
- At what granularity should puzzles be applied?
  - Per byte(s)?
  - Per packet(s)?
  - Per flow(s)?
  - Per flow aggregate?
  - Driven by performance and level of protection required

Control

- Puzzles require control algorithms to maintain high utilization and low loss
  - Mandatory, multi-resolution ECN signals that can be given at any time granularity
  - Can apply ideas from TCP/AQM control
    - Adapt puzzle difficulty within network based on load
    - Adapt end-host response to maximize throughput while minimizing system resource consumption (natural game theoretic operation)
  - Hypothesis
    - Easier to design puzzle controllers versus those used in TCP/AQM

Fairness

- Mechanism in a layer with minimal information
  - Can support bandwidth-based puzzle delivery
  - Can support some differentiation to deter Smurf/Fraggle
- But.. 202.183.197.116 - - [02/Jun/2003:02:08:29 -0700] "GET /default.ida?XXX%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0" 404 306 "-" "-"
- Need a "puzzle manager" to drive IP-layer puzzle policy based on application input



Fairness

- Enables "Reputation-based networking"
  - Software vendors
    - Making "trustworthy computing" mandatory (not marketing)
    - Long-term, computational tax for poorly designed software
  - System administrators and IT practices
    - Making responsible system management mandatory
    - Disturbing pervading notion: "cheaper to leave infected than patch"
    - Long-term, computational tax on poorly administered systems
  - End-users
    - Making users choose more secure software and adopt better practices
    - Punish users behaving "badly"
    - Long-term, computational tax on ignorance and maliciousness
  - "Nothing is certain but death and taxes."

Fairness

- Inserting a “trust” estimator into the knowledge plane
  - Answer the “WHO” question?
    - Who is a likely source of a future DoS attack?
  - No keys, no signatures, no centralized source
  - Based on time-varying distributed view of client behavior
  - Similar to GeoNetMap's “confidence” measure

Deployment

- This space left intentionally blank

Deployment

- Can be transparently and incrementally deployed via puzzle firewalls/proxies
- Application-driven puzzle manager requires more intrusive changes
- Financial incentive to change is present
  - Lost productivity (see last two weeks)
  - Lost revenue, services (WWW, power, ATM, etc.)
    - SoBig.* author laughing all the way to the bank (Grrrr....)
  - Change may need a kick from the government?
- If not, RIPIPP$_{uzzles}$ 2020?
  - Putting in early...

Status

- User-level UDP forwarder
  - Tamper-proof operation (must be along path to deny service)
  - Puzzle generation ~1µs
  - Puzzle verification ~1µs, constant amount of state
  - Fine-grained puzzle difficulty adjustment
  - 20,000 puzzles/sec on commodity hardware
  - 250Mbs+ for per-packet puzzles with MTU packets
  - Small packet overhead
    - Puzzle question ~40 bytes
    - Puzzle answer ~20 bytes
- Currently working on native IP/ICMP implementation
  - `netfilter/iptables`
  - Puzzle-protected Counter-strike through puzzle firewall/proxy

Questions?

- PuzzleNet and Reputation-based Networking

http://www.cse.ogi.edu/sysl/projects/puzzles