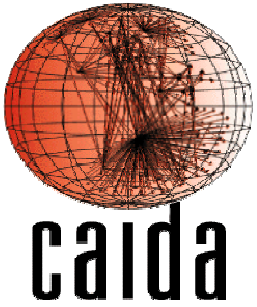
Internet Worms:
Current Capabilities in Awareness, Detection, Response

Colleen Shannon (CAIDA)
David Moore (CAIDA/UCSD-CSE)

cshannon @ caida.org
dmoore @ caida.org
www.caida.org



Outline

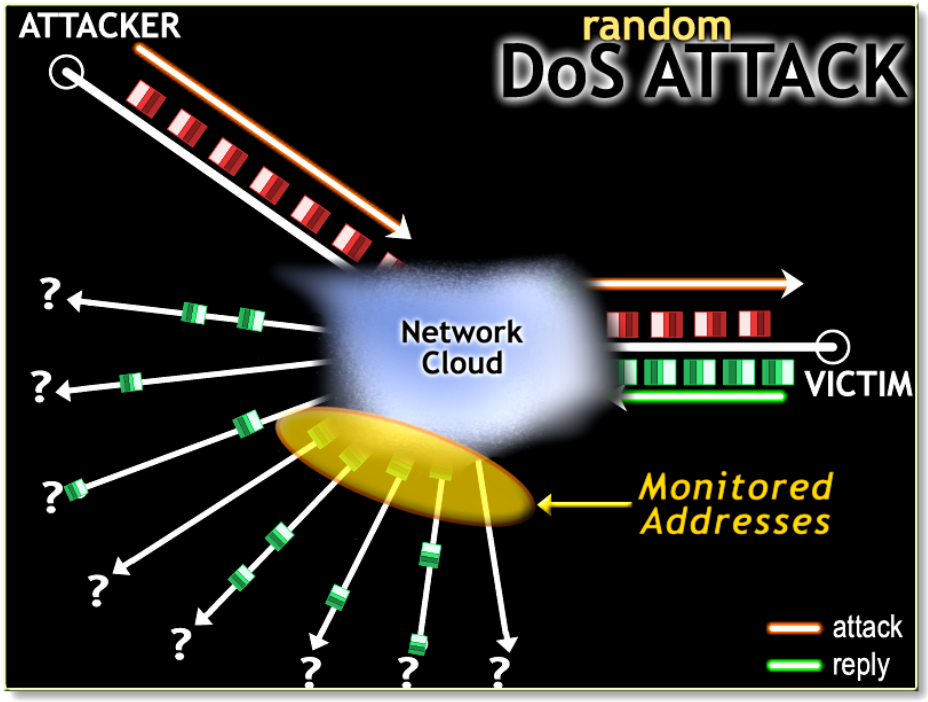
- Detecting Internet Worms
  - Network Telescope
- Recent Internet Worms
  - Code Red
  - SQL Slammer (Sapphire)
- Worm Quarantine
  - How well could it work?

Network Telescope

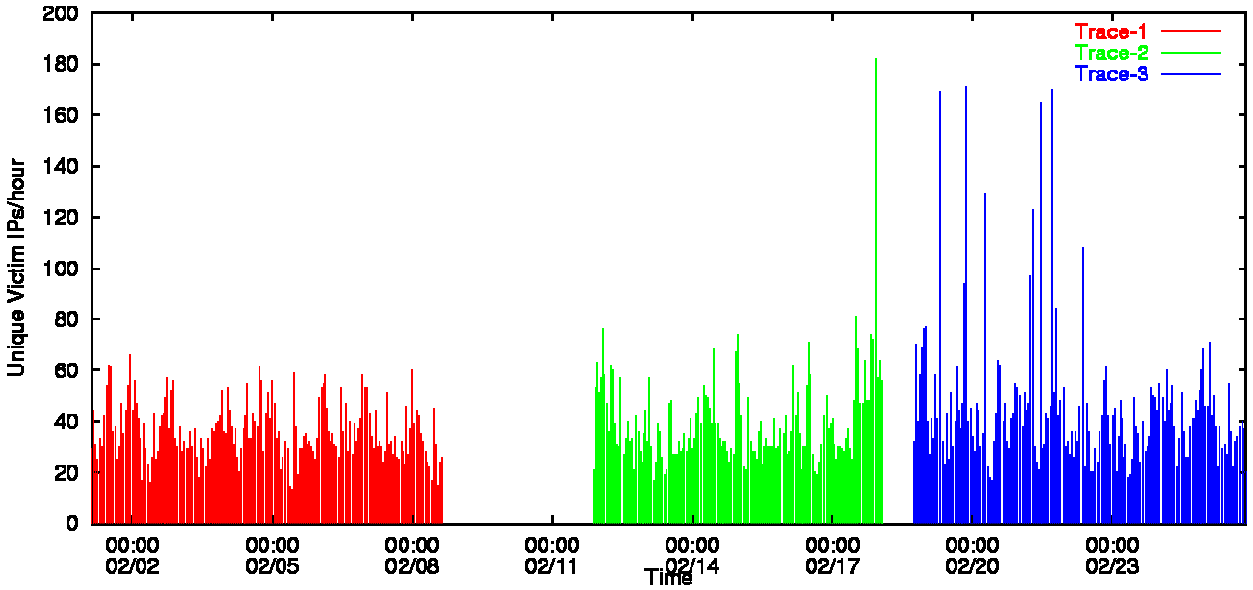
- Chunk of (globally) routed IP address space
  - 16 million IP addresses
- Little or no legitimate traffic (or easily filtered)
- Unexpected traffic arriving at the network telescope can imply remote network/security events
- Generally good for seeing explosions, not small events
- Depends on random component in spread

Network Telescope: Denial-of-Service Attacks

- Attacker floods the victim with requests using random spoofed source IP addresses
- Victim believes requests are legitimate and responds to each spoofed address
- We observe 1/256th of all *victim responses* to spoofed addresses [MSV01]



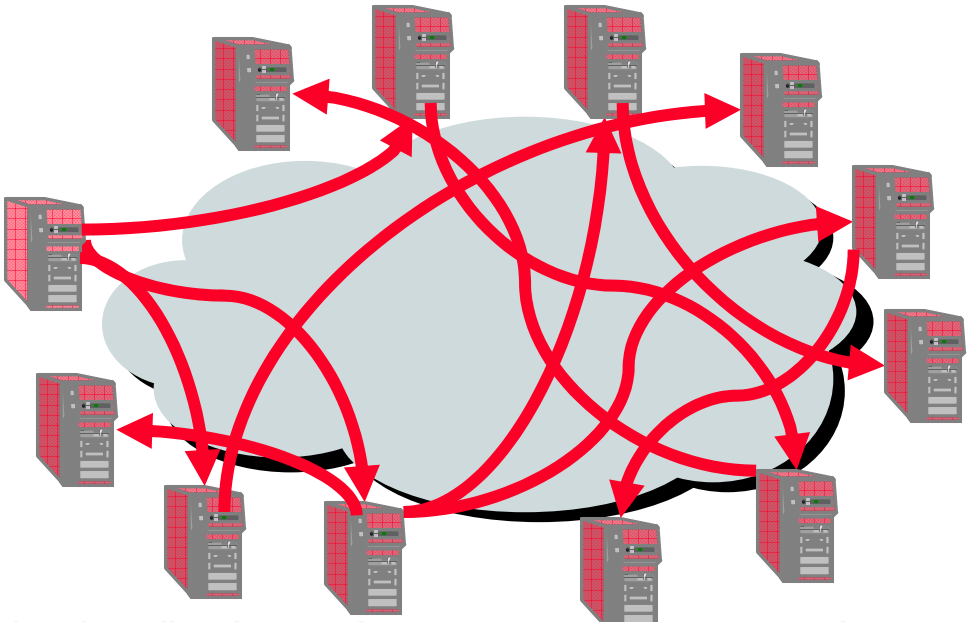
Denial-of-Service Attacks



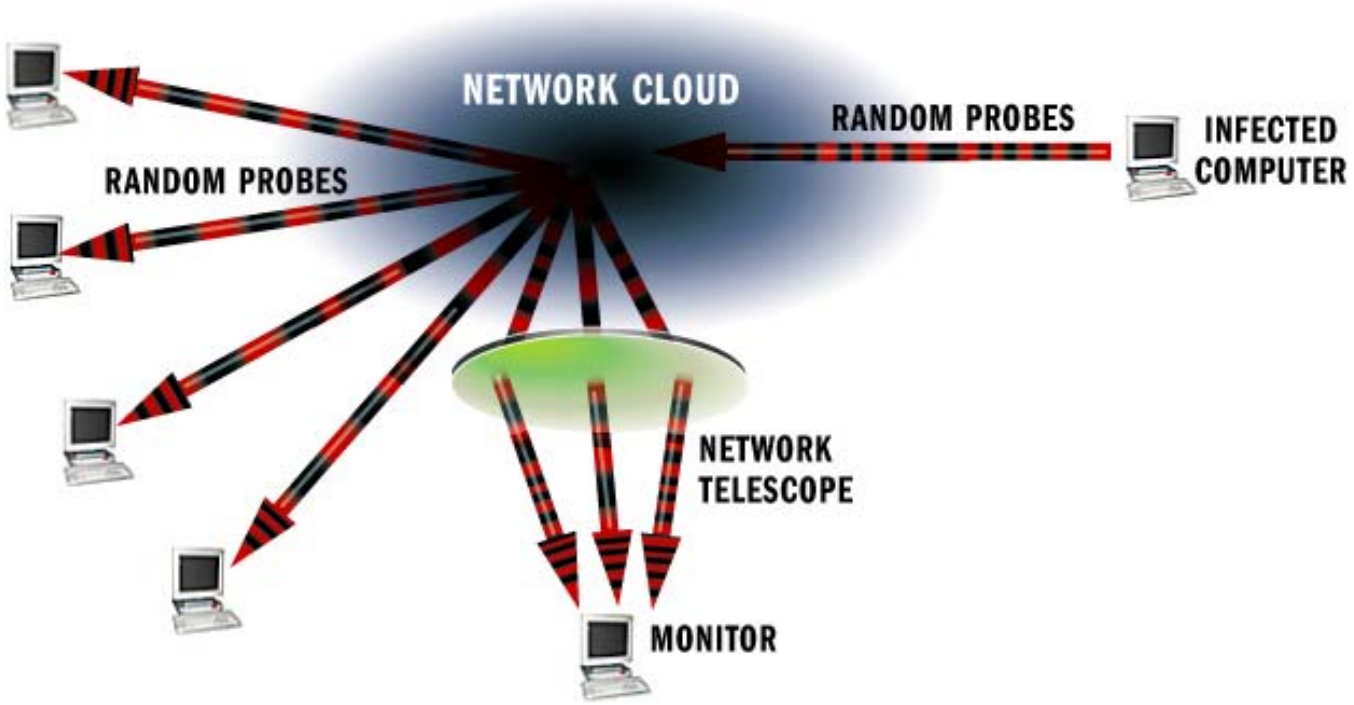
- Current denial-of-service statistics:
  - 300 ongoing denial-of-service attacks every minute
  - 890 unique victims per day
  - 3481 denial-of-service attacks per week

What is a Network Worm?

- Self-propagating self-replicating network program
  - Exploits some vulnerability to infect remote machines
    - No human intervention necessary
  - Infected machines continue propagating infection



Network Telescope: Worm Attacks



- Infected host scans for other vulnerable hosts by randomly generating IP addresses
- We monitor 1/256th of all IPv4 addresses
- We see 1/256th of all worm traffic of worms with no bias and no bugs

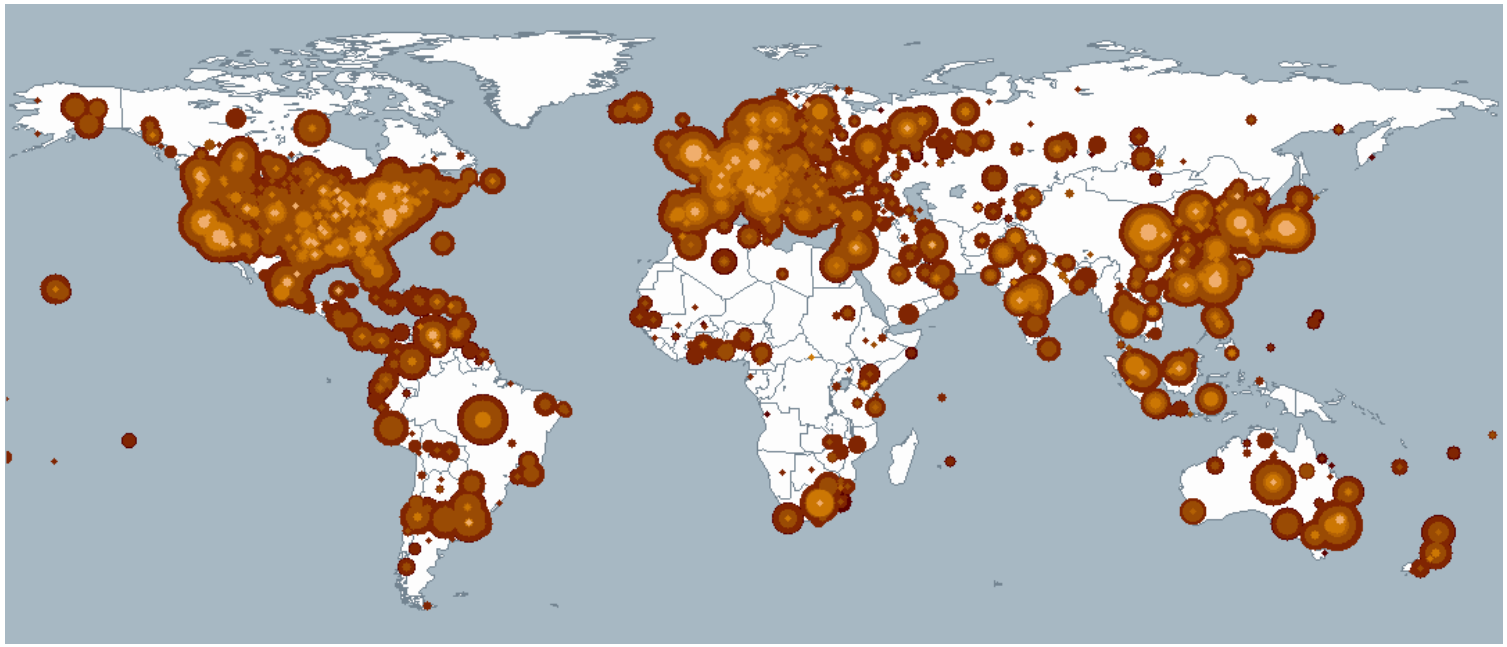
Internet Worm Attacks: Code-Red

(July 19, 2001)

- Animation

Internet Worm Attacks: Code-Red

(July 19, 2001)



- 360,000 hosts infected in *ten hours*
- No effective patching response
- More than $1.2 billion in economic damage in the first ten days
- Collateral damage: printers, routers, network traffic

Response to August 1st CodeRed

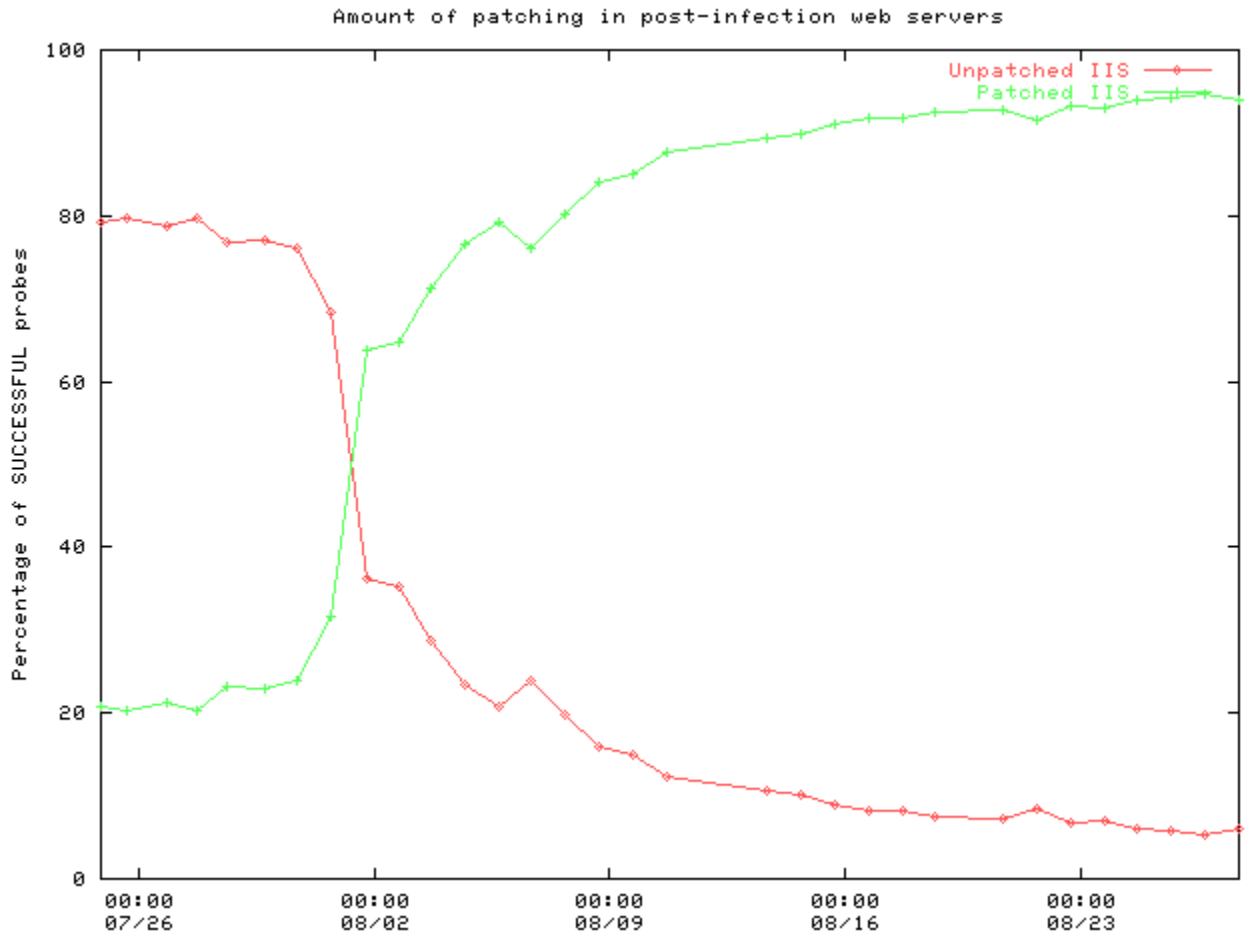
- CodeRed was programmed to deactivate on July 20th and begin spreading again on August 1st
- By July 30th and 31st, more news coverage than you can shake a stick at:
  - FBI/NIPC press release
  - Local ABC, CBS, NBC, FOX, WB, UPN coverage in many areas
  - National coverage on ABC, CBS, NBC, CNN
  - Printed/online news had been covering it since the 19th
- "Everyone" knew it was coming back on the 1st

- Best case for human response: known exploit with a viable patch and a known start date

Patching Survey

- How well did we respond to a best case scenario?
- Idea: randomly test subset of previously infected IP addresses to see if they have been patched or are still vulnerable
- 360,000 IP addresses in pool from initial July 19th infection
- 10,000 chosen randomly each day and surveyed between 9am and 5pm PDT

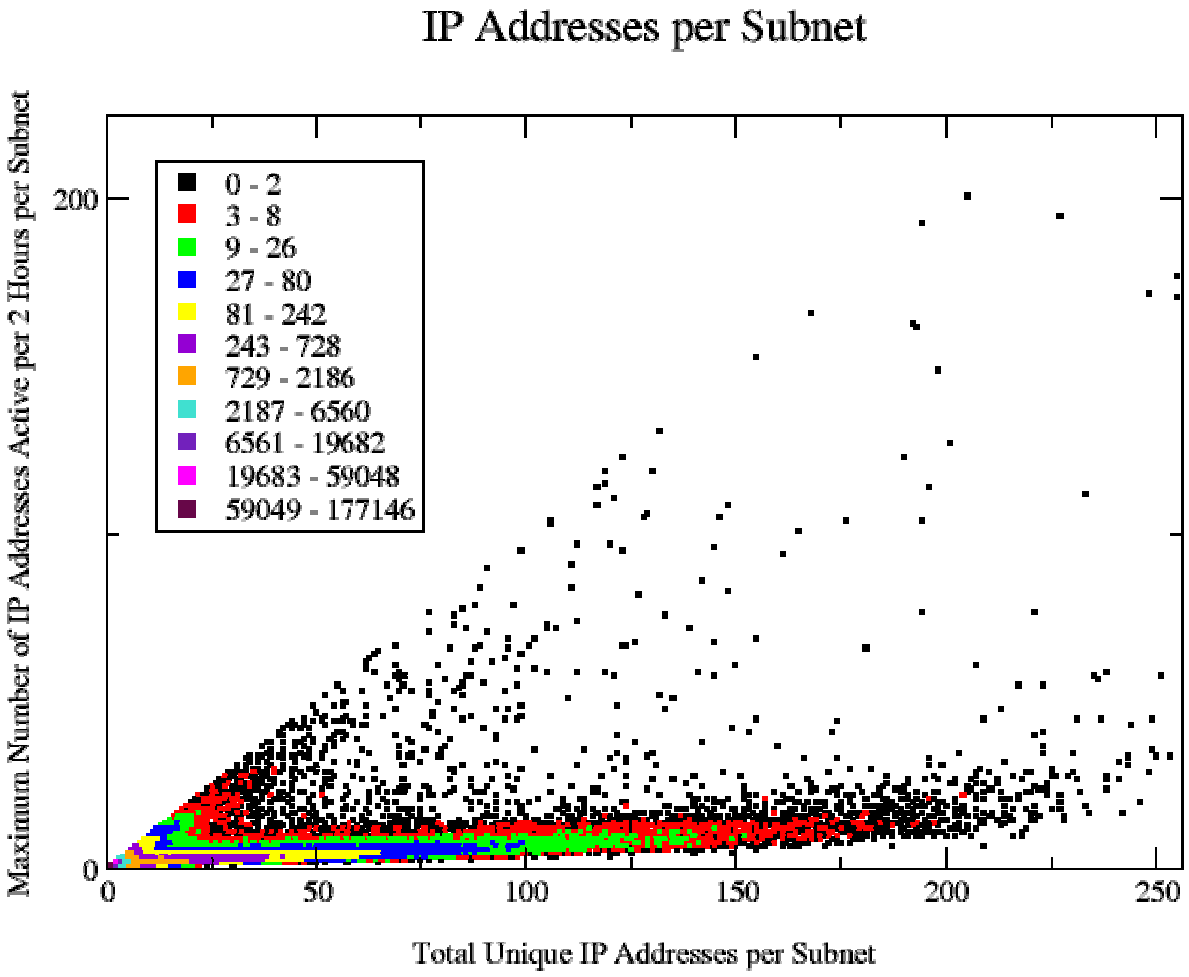
Patching Rate



Dynamic IP Addresses

- How can we tell how when an IP address represents an infected **computer**?
- Resurgence of CodeRed: Max of ~180,000 unique IPs seen in any 2 hour period, but more than 2 million across ~a week.
- This ***DHCP effect*** can produce skewed statistics for certain measures, especially over long time periods

DHCP Effect seen in /24s



Summary of Recent Events

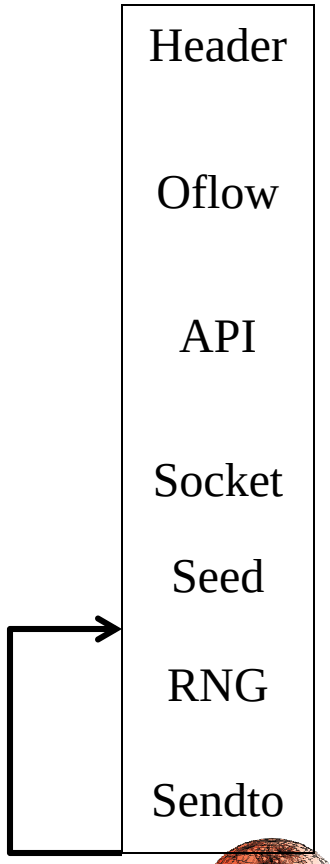
- **CodeRed** worm released in Summer 2001
  - Exploited buffer overflow in IIS
  - Uniform random target selection (after fixed bug in CRv1)
  - Infects 360,000 hosts in 10 hours (CRv2)
  - Still going…
- Starts **renaissance** in worm development
  - CodeRed II
  - Nimda
  - Scalper, Slapper, Cheese, etc.
- Culminating in **Sapphire/Slammer** worm (Winter 2003)

Inside the Sapphire/Slammer Worm

- Exploited bug in MSSQL 2000 and MSDE 2000
- Worm fit in a single UDP packet (404 bytes)

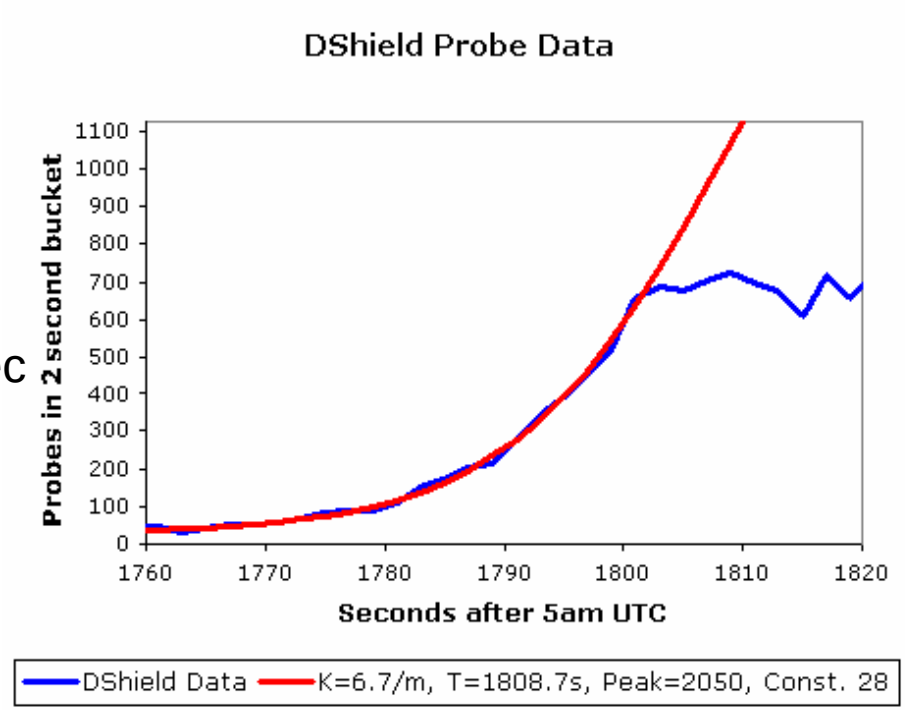
- Simple code structure } Code borrowed from published exploit
  - Cleanup from buffer overflow
  - Get API pointers

  - Create socket & packet
  - Seed RNG with `getTickCount()`
  - While (TRUE)
    - Increment RNG (mildly buggy)
    - Send packet to RNG address
- Key insight: non-blocking & stateless scanning (adaptable to TCP-based worms)

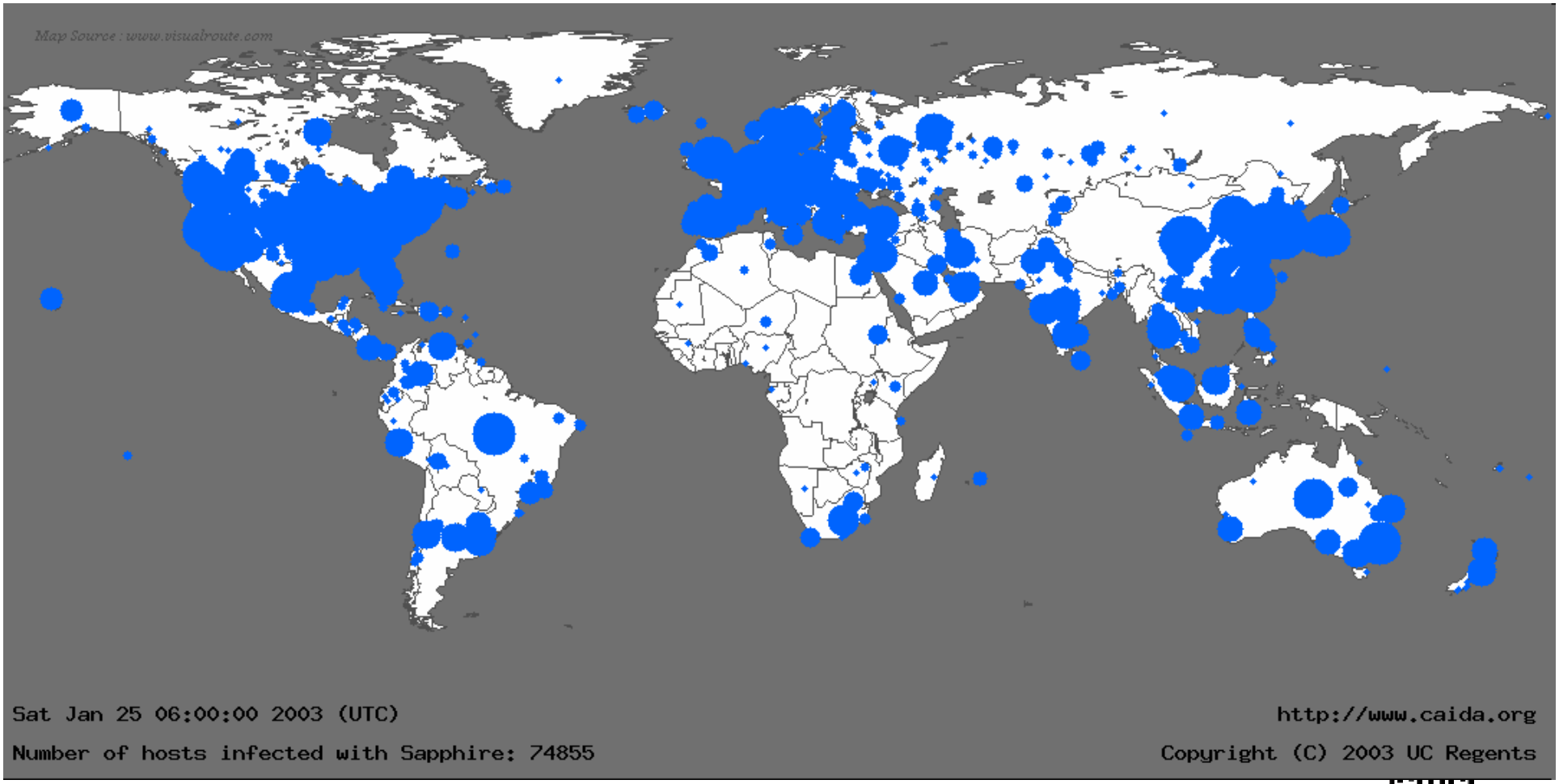


Sapphire growth

- First ~1min behaves like classic random scanning worm
  - Doubling time of ~8.5 seconds
  - Code Red doubled every 40mins
- >1min worm starts to saturate access bandwidth
  - Some hosts issue >20,000 scans/sec
  - Self-interfering
- Peaks at ~3min
  - 55million IP scans/sec
- 90% of Internet scanned in <10mins
  - Infected ~100k hosts (conservative due to PRNG errors)

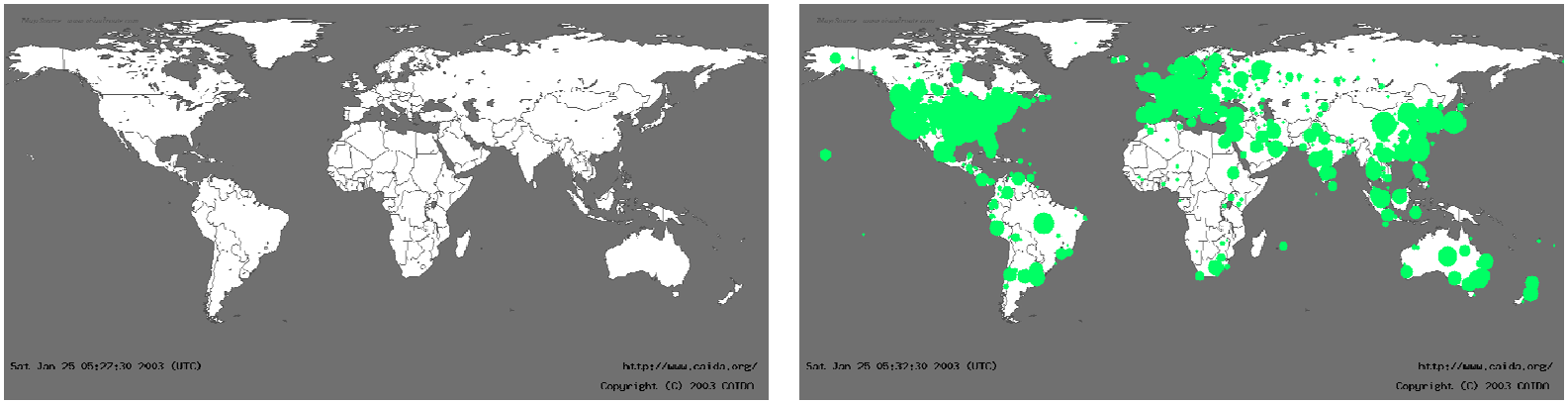


Sapphire Animation



Internet Worm Attacks: Sapphire

(aka SQL Slammer) – Jan 24, 2003



Before 9:30PM (PST) After 9:40PM (PST)

- Over 75,000 hosts infected in *ten **minutes***
- Sent more than 55 million probes per second world wide
- Collateral damage: Bank of America ATMs, 911 disruptions, Continental Airlines cancelled flights
- Unstoppable; relatively benign to hosts

The Sky is Falling…

- **Worms are the worst Internet threat today**
  - Many *millions* of susceptible hosts
  - *Easy* to write worms
    - Worm payload separate from vulnerability exploit
    - Significant code reuse in practice
  - Possible to cause major damage
    - Lucky so far; existing worms have benign payload
    - Wipe disk; flash bios; modify data; reveal data; Internet DoS
- **We have no operational defense**
  - Good evidence that humans don't react fast enough
  - Defensive technology is nascent at best

What can we do?

- **Measurement**
  - What are worms doing?
  - What types of hosts are infected?
  - Are new defense mechanisms working?
- **Develop operational defense**
  - Can we build an automated system to stop worms?

Network Telescope Observation Station

- Continuous data collection with rotating data files:
  - full packet trace kept for 24 hours
  - complete packet header trace kept for 1 week
  - aggregated data (e.g. flow tables) stored indefinitely
- Sanitized data publicly accessible

- Eventual expansion to include monitoring distributed address space

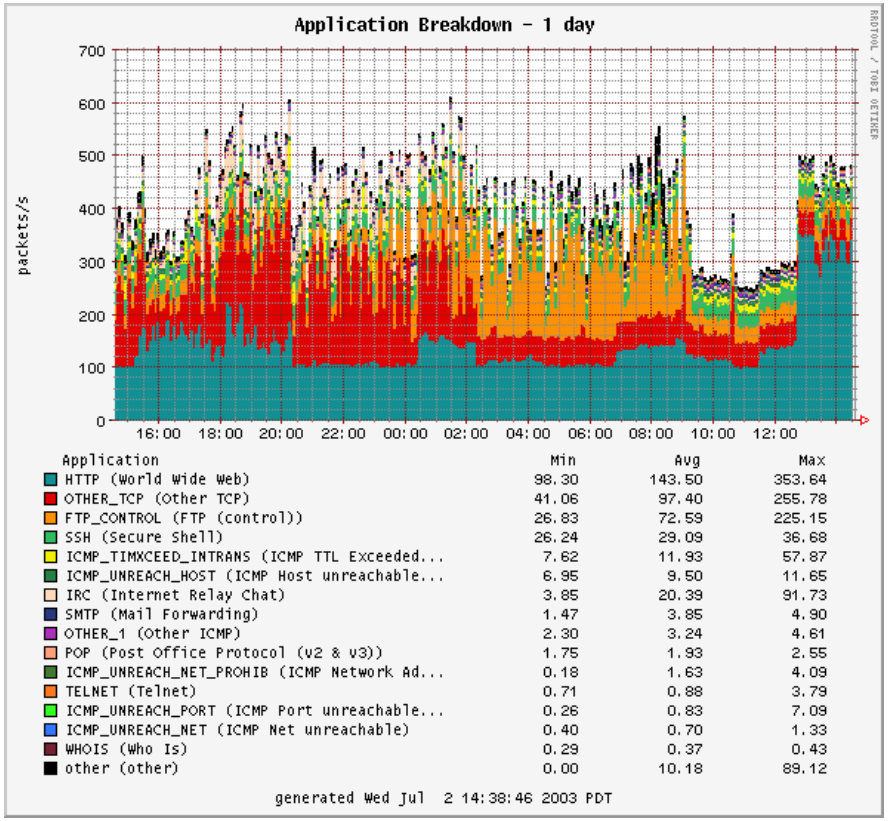
- Planned data collection/display system – does not yet exist

NTOS Graphical Interface

- Publicly accessible realtime graphical monitor
  - denial-of-service attacks
  - worm activity
  - port scanning
- Authorized users:
  - Drilldown functionality:
    - time scale
    - transport protocol
    - application ports
  - Ability to save (manually or automatically) data of interest
  - Email/pager alerts for trigger events

NTOS Graphical Interface:
Global Backscatter Traffic

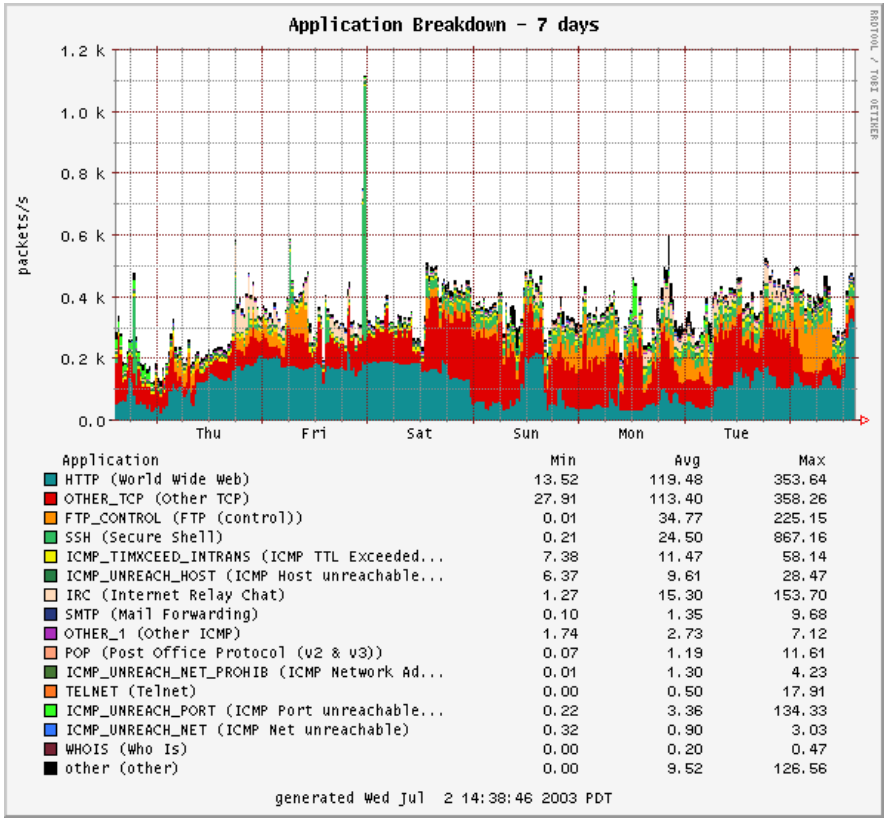
- July 1, 2003
- Backscatter across a day highly variable
- Continuous port 80 attacks
- Intermittent FTP attacks
- Intermittent IRC attacks (often classified as “Other TCP”)



NTOS Graphical Interface:
Global Backscatter Traffic

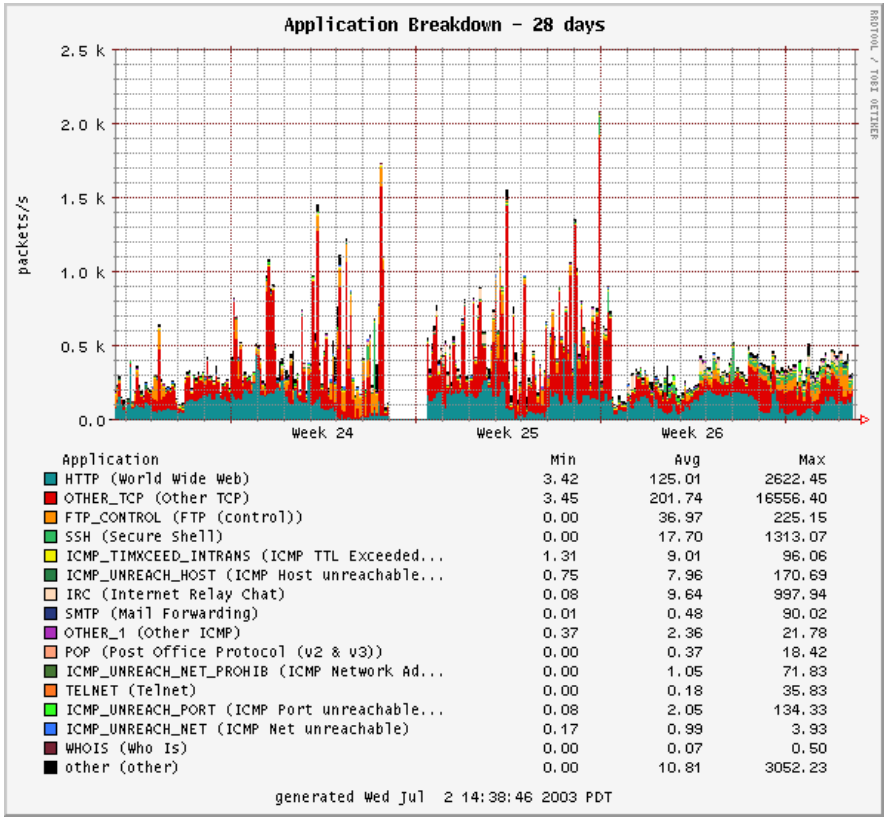
- June 25 - July 1, 2003 (one week)

- Traffic level highly variable
- Some very large volume attacks
- Some attacks missed because traffic volume crashed monitor



NTOS Graphical Interface:
Global Backscatter Traffic

- July 1, 2003
- Continuous ~300k packet/second backscatter
- Intermittent large attacks up to ~20k packets/second
- Huge traffic influx overloads monitor

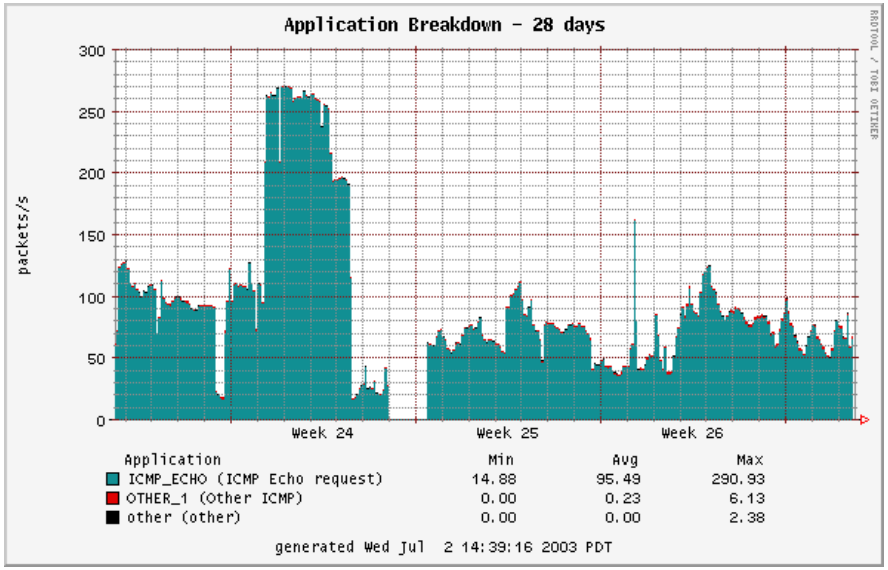


NTOS Graphical Interface:
Global Host Scanning

- ICMP Host Scanning
- June 2003

- Blue bars show sapphire traffic from a single host

- Huge traffic influx overloaded monitor

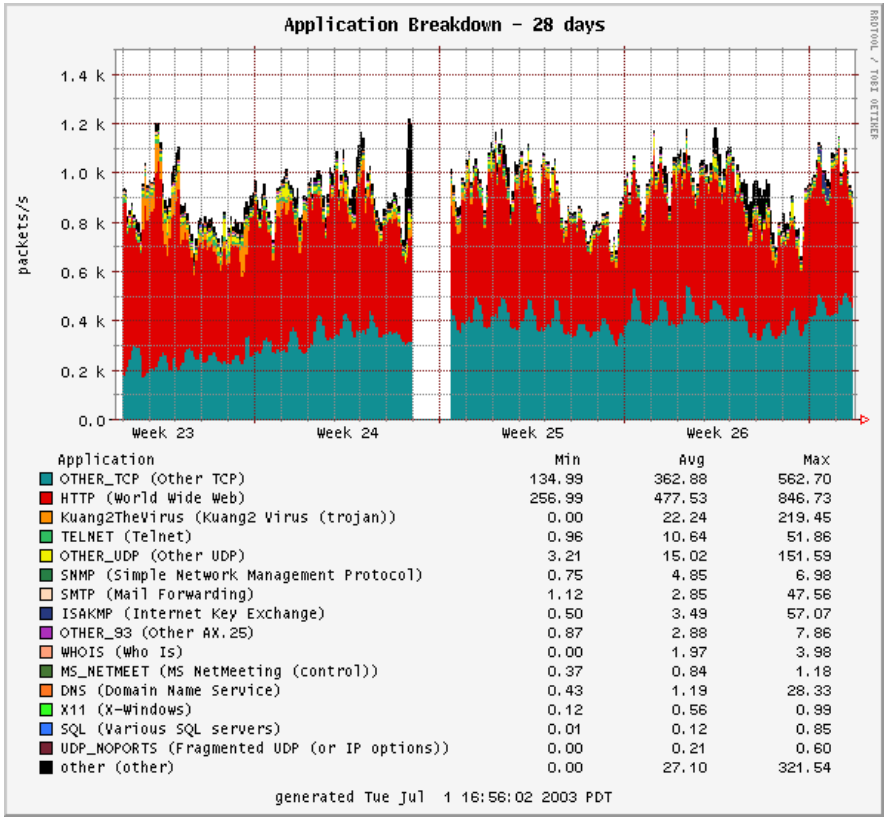


NTOS Graphical Interface:
Global Worm/Scan Traffic

- Worm / Port Scan Traffic
- June 2003

- Blue bars show sapphire traffic from a single host

- Huge traffic influx overloaded monitor



Summary

- Worms are the worst threat to the Internet today
  - Millions of remotely exploitable bugs
  - Millions of unpatched machines
  - Fast worms are easy to write
  - Only a matter of time before we see a malicious payload
- Planned Network Telescope Observation Station:
  - Realtime monitor of:
    - worm spread
    - denial-of-service
    - worm and port scans
  - Archived data for in-depth analysis.

On to Part 2…

Is it possible to stop Internet worms?

Open Research Questions

- Denial-of-Service Attacks
  - interactive timeouts
  - multiple protocol attacks
  - multiple attacks against a single victim
  - overall trends
- Internet Worms
  - random number generation and spread rates
  - victim classification/hitlists
  - effective countermeasures

Acknowledgements

- Collaborators:
  - UCSD-CSE: Geoff Voelker, Stefan Savage, Jeffrey Brown
  - ICSI: Vern Paxson
  - Silicon Defense: Stuart Staniford, Nicholas Weaver
  - UCB-CSE: Nicholas Weaver

- Data Providers:
  - UCSD: Brian Kantor, Pat Wilson
  - UCB/LBL: Vern Paxson
  - UWISC: Dave Plonka
  - Dshield: Johannes Ullrich
  - Compaq/WRL: Jeff Mogul
  - DOD CERT: Donald LaDieu, Matthew Swaar

- Funding:
  - Cisco University Research Program (URP)
  - DARPA
  - NSF
  - CAIDA Members

Related Papers

- Inferring Internet Denial-of-Service Activity [MSV01]
  - David Moore, Stefan Savage, Geoff Voelker
  - http://www.caida.org/outreach/papers/2001/BackScatter/
- Code-Red: A Case Study on the spread and victims of an Internet Worm [MSB02]
  - David Moore, Colleen Shannon, Jeffrey Brown
  - http://www.caida.org/outreach/papers/2002/codered/
- Internet Quarantine: Requirements for Containing Self-Propagating Code [MSVS03]
  - David Moore, Colleen Shannon, Geoff Voelker, Stefan Savage
  - http://www.caida.org/outreach/papers/2003/quarantine/
- The Spread of the Sapphire/Slammer Worm [MPS03]
  - David Moore, Vern Paxson, Stefan Savage, Colleen Shannon, Stuart Staniford, Nicholas Weaver
  - http://www.caida.org/outreach/papers/2003/sapphire/

Reference

- Code-Redv1, Code-Redv2, CodeRedII, Nimda
  - http://www.caida.org/analysis/security/code-red/
- Code-Redv2 In-depth analysis
  - http://www.caida.org/analysis/security/code-red/coderedv2_analysis.xml
- Spread of the Sapphire/SQL Slammer Worm
  - http://www.caida.org/analysis/security/sapphire/