Defending Computer Networks
Lecture 8: More Port Scanning and Worms

Stuart Staniford

Adjunct Professor of Computer Science

Logistics

- HW2 hopefully out later today
- Quiz next time
  - Twenty minutes
  - No notes/laptops/tablets/phones/etc
- Z taking lecture next Thursday

Is Wall Street ready for the next cyberattack?

Anita Balakrishnan

13 Hours Ago

CNBC

Take a deep breath, and imagine a doomsday scenario on Wall Street: a hacktivist group coordinates a large-scale, three-day attack on the capital markets meant to disrupt trading and confidence in the U.S. markets.

That's what cybersecurity firm SIFMA tried to simulate in a Wednesday experiment—where they found that banks might be limited during a hacking by laws that restrict information sharing.

> The third iteration of the experiment, called Quantum Dawn 3, included 80 financial institutions and government entities, such as the Department of the Treasury and FBI. It started with smaller-scale attacks targeted at individual entities through multiple types of systems, followed by a larger attacks that interrupt market operations.

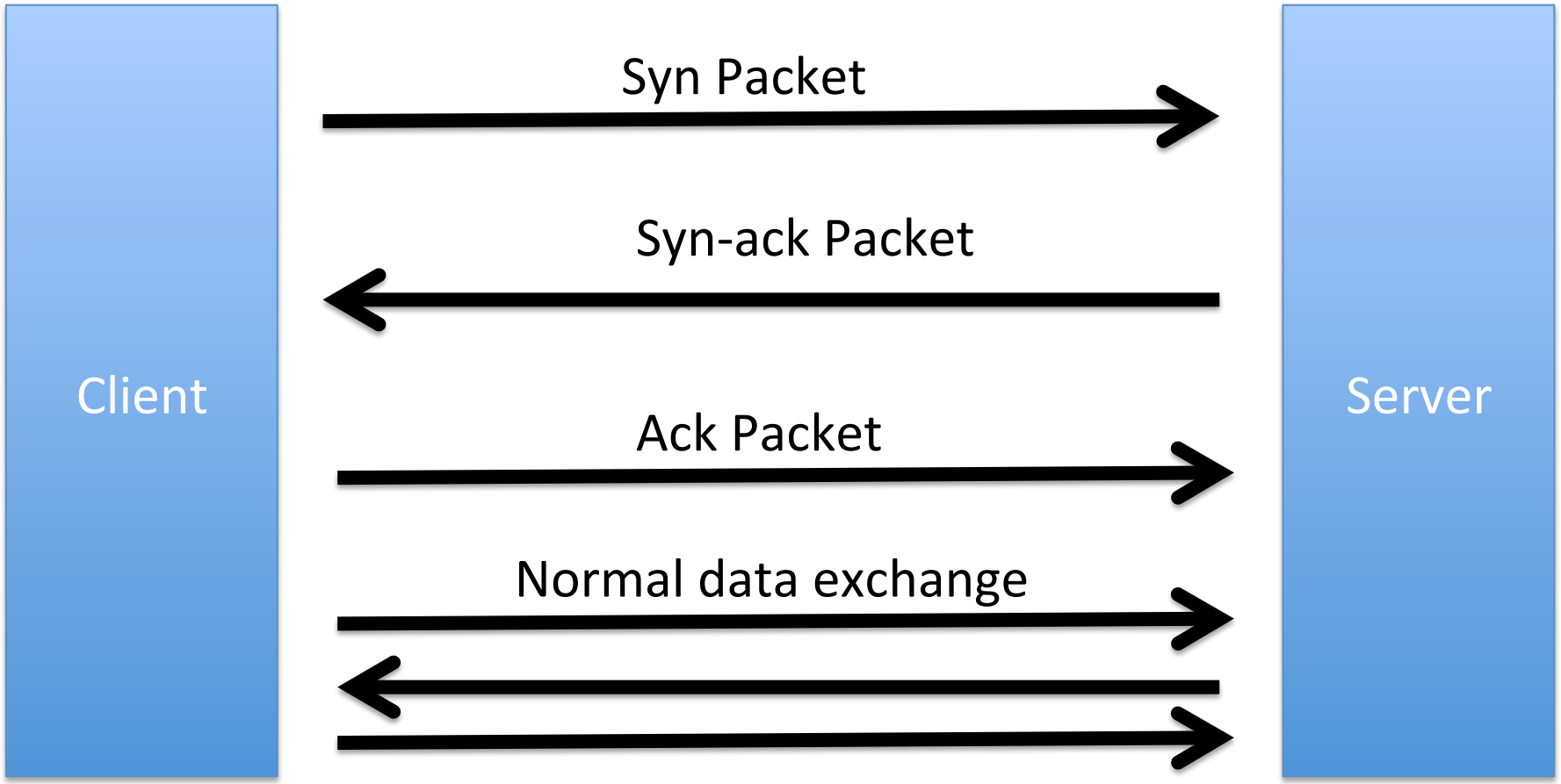
New Assigned Reading

- Staniford et al *Practical Automated Detection of Stealthy Portscans* http://webpages.cs.luc.edu/~pld/courses/intrusion/fall05/hoagland_spade.pdf
  - Through section 3.1
- Staniford et al *How to 0wn the Internet in Your Spare Time*. http://www.icir.org/vern/papers/cdc-usenix-sec02/
- Falliere et al. *W32.Stuxnet Dossier*. http://www2.gwu.edu/~nsarchiv/NSAEBB/NSAEBB424/docs/Cyber-044.pdf

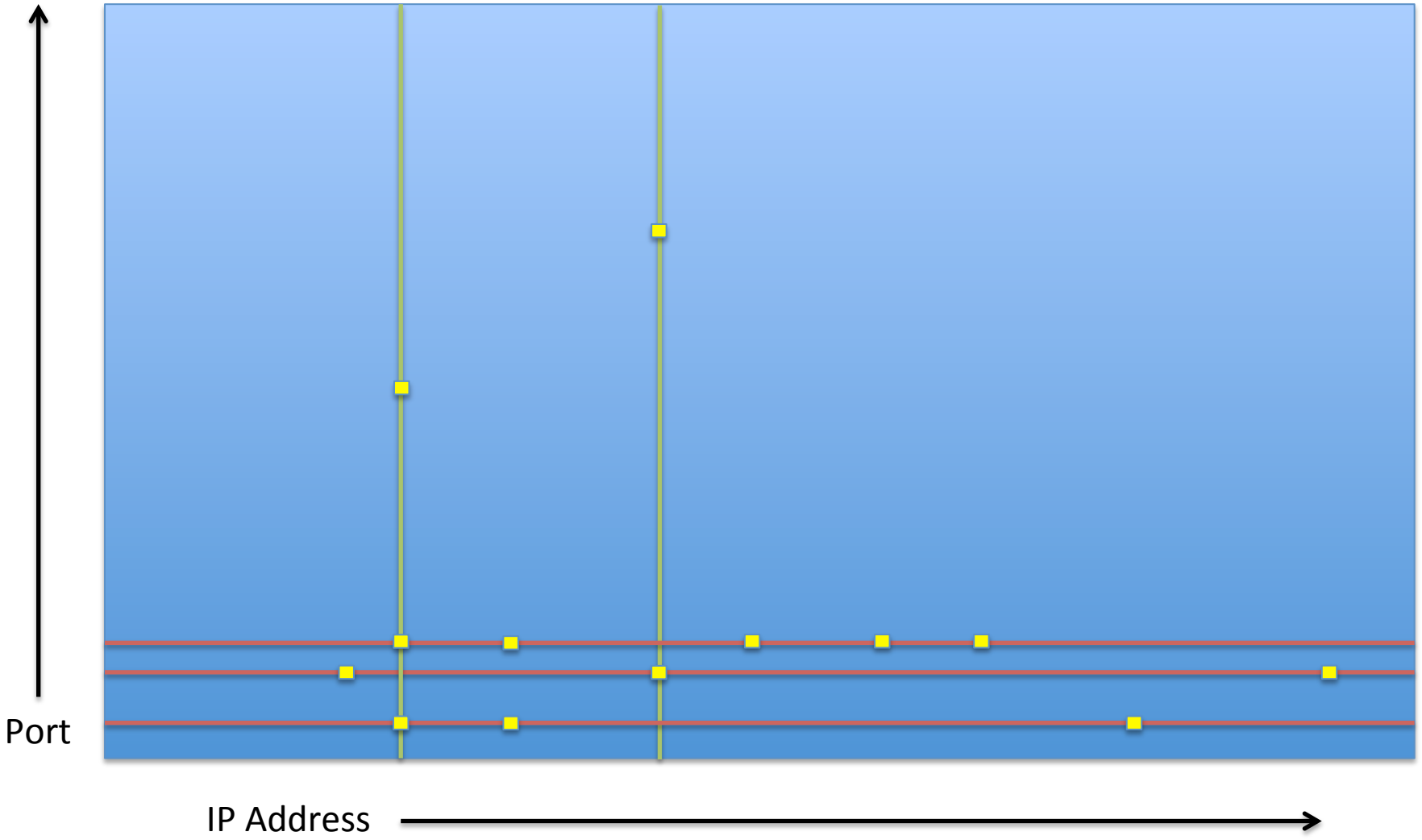
Main Goals for Today

- Finish up portscanning/detection
- Worms

Refresh: 3-way handshake

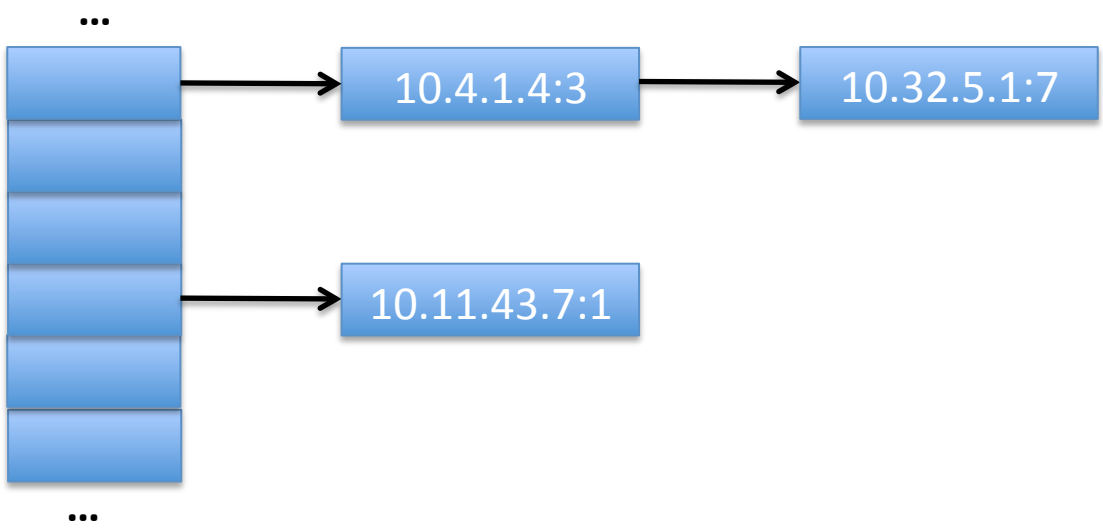


Visualizing Scans



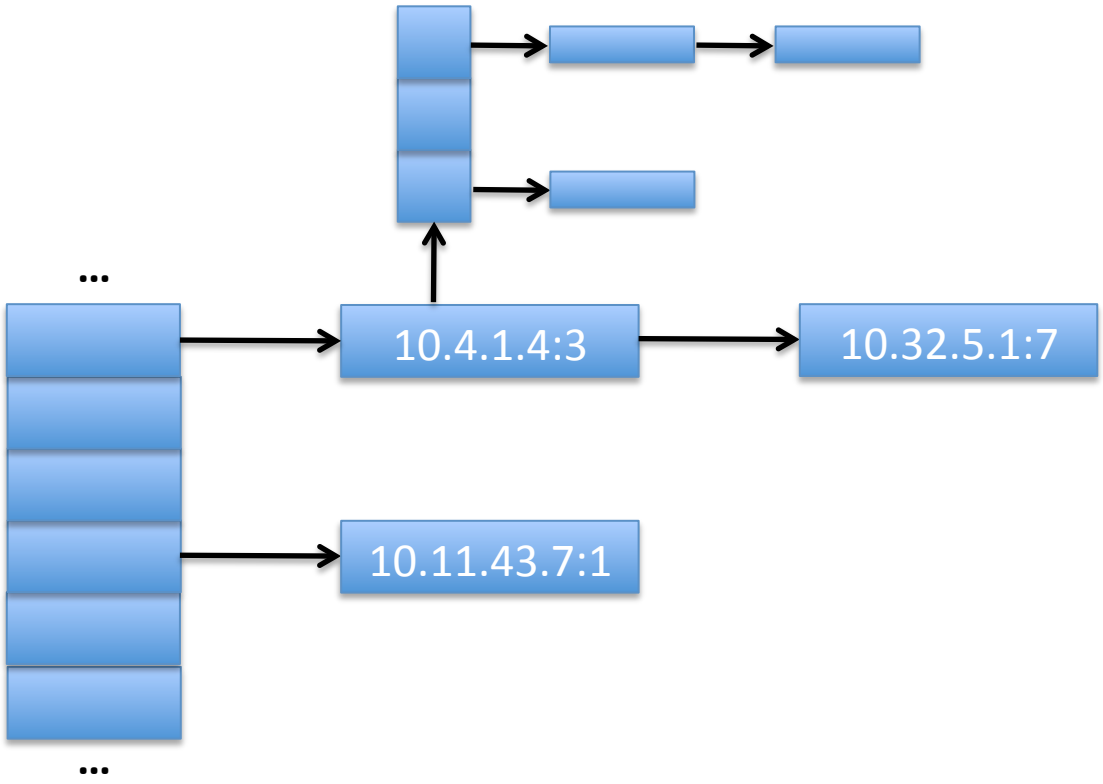
Then we need a data structure

- Simplest possible thing is a hash table
  - keyed on client IP
  - With per-connection counts of relevant stuff
  - Eg just count syns
  - Portscanners will issue more syns than average.
    - Alert when count goes over threshold
    - But what's likely to go wrong?



Keep track of unique dests/src?

- Now have to have a way to know
  - what is a unique dst for that src?



Better Idea

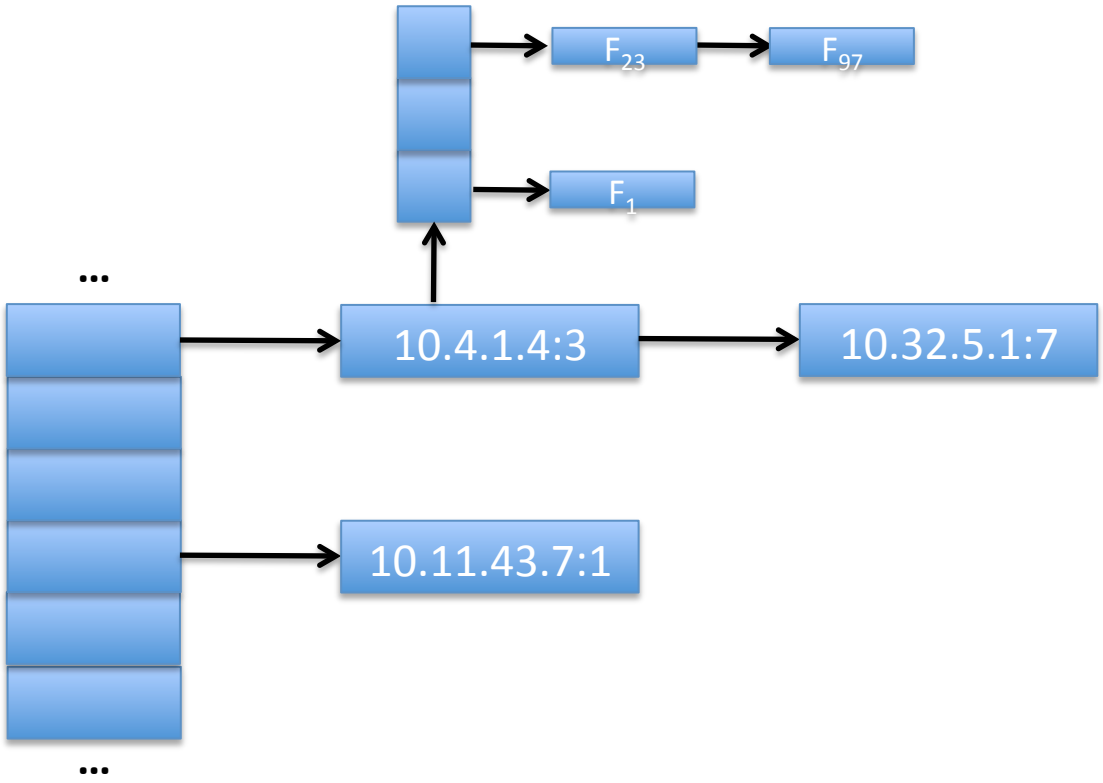
- Key off the idea that port-scanners make a lot of failed connections.
- Legit users make only a few
  - So keep track of "failed-succeed" count
  - Alert when goes over threshold.
- How can the attacker game this?
- Doesn't work in the presence of packet-filter/ firewalls.

Another Idea

- Learn the probability of a syn (say) being to a destination:
  - P(D)
  - Popular servers will have high P(D) (say 5% or 1%)
  - Non-servers will have very low P(D) (1 in 10^6 or 10^9)
  - Take –log(P(D)) and accumulate *that* in hash table
    - Anomaly score
  - Portscanners will accumulate a lot of anomaly score
    - Alert if over a threshold
  - Harder for attackers to game – don't know P(D)
    - Otherwise wouldn't need to portscan

Extending the basic idea

- Keep flow table state
- Know when we see things like unexpected F
- Give that a high anomaly score



Cabinet ministers' email hacked by Isil spies

Intelligence agency investigation discovers extremists linked to the Islamic State of Iraq and the Levant (Isil) have been targeting information held by some of David Cameron's most senior ministers

Jihadists in Syria have hacked into ministerial email accounts in a sophisticated espionage operation uncovered by GCHQ, the Telegraph can disclose.

An investigation by the intelligence agency has discovered that extremists linked to the Islamic State of Iraq and the Levant (Isil) have been targeting information held by some of David Cameron's most senior ministers, including Theresa May, the Home Secretary.

http://www.telegraph.co.uk/news/politics/11859005/Cabinet-ministers-email-hacked-by-Isil-spies.html

Definitions

- Worm: A malicious program capable of infecting and running on other systems across the network via security weaknesses in that system in order to spread.
- Virus: Malicious code capable of infecting and attaching itself to other executables in order to spread.
  - Was more important in the PC/floppy drive era.

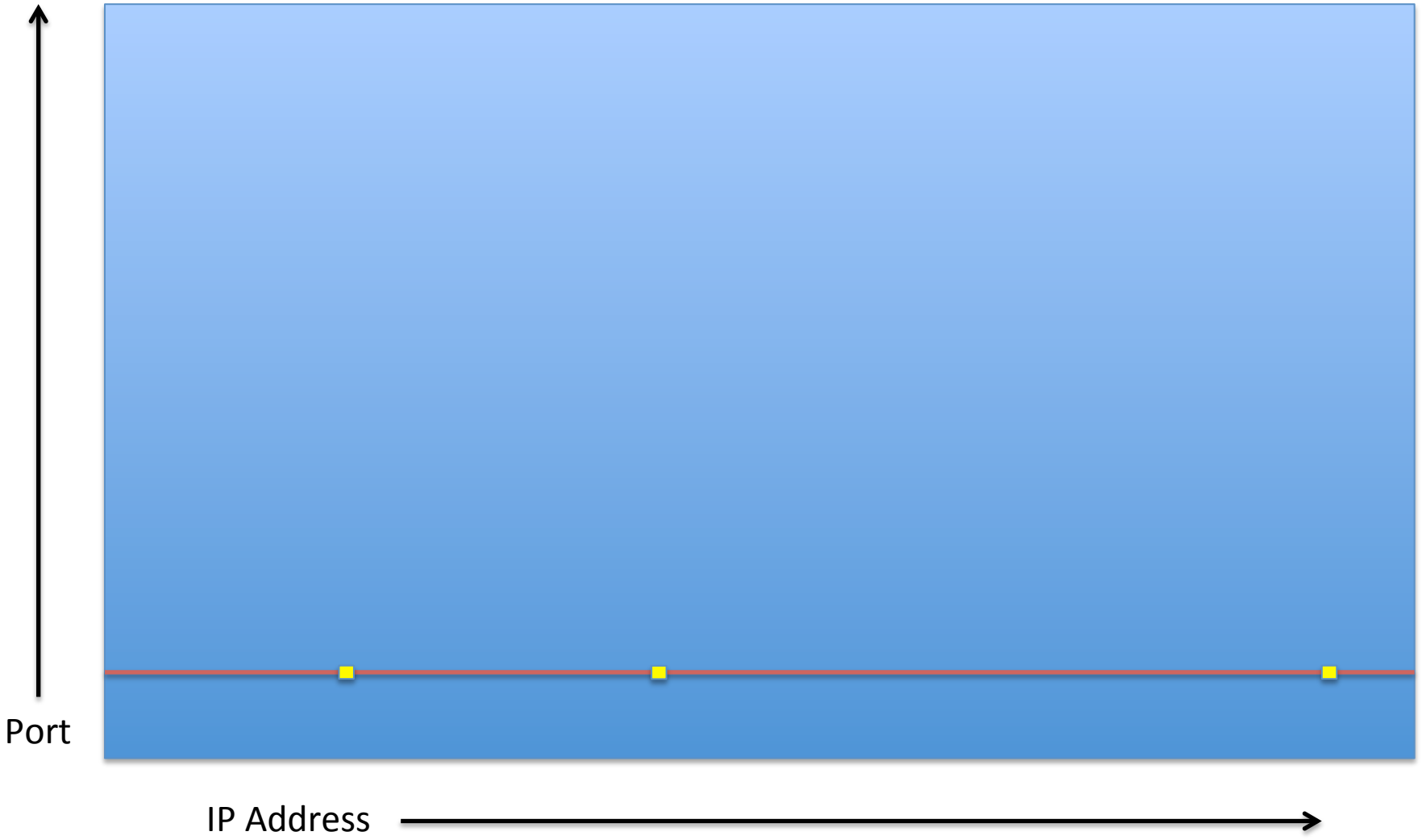
Ancient History

- 1988 Internet Worm (first serious worm problem)
  - Infected BSD Unix systems
  - Robert Morris Jr – Cornell student at the time.
  - Exploited vulnerabilities
    - Stack overflow in finger
    - Unauthenticated debug functionality in sendmail
    - guessing weak passwords.
  - Looked on host for info about other hosts.
    - Topological worm.
  - "disrupted normal activities and Internet connectivity for many days"
    - http://spaf.cerias.purdue.edu/tech-reps/823.pdf

Worms

- If we can scan,…
- And we have an exploit for a server port
- We have the basis for a worm!

Random Scanning Worms



Scanning Worms Big in the early 2000s

- 2001 Code Red (I and II), Nimda
- 2003 Slammer and Blaster/Welchia
- 2004 Sasser
- 2008 Conficker

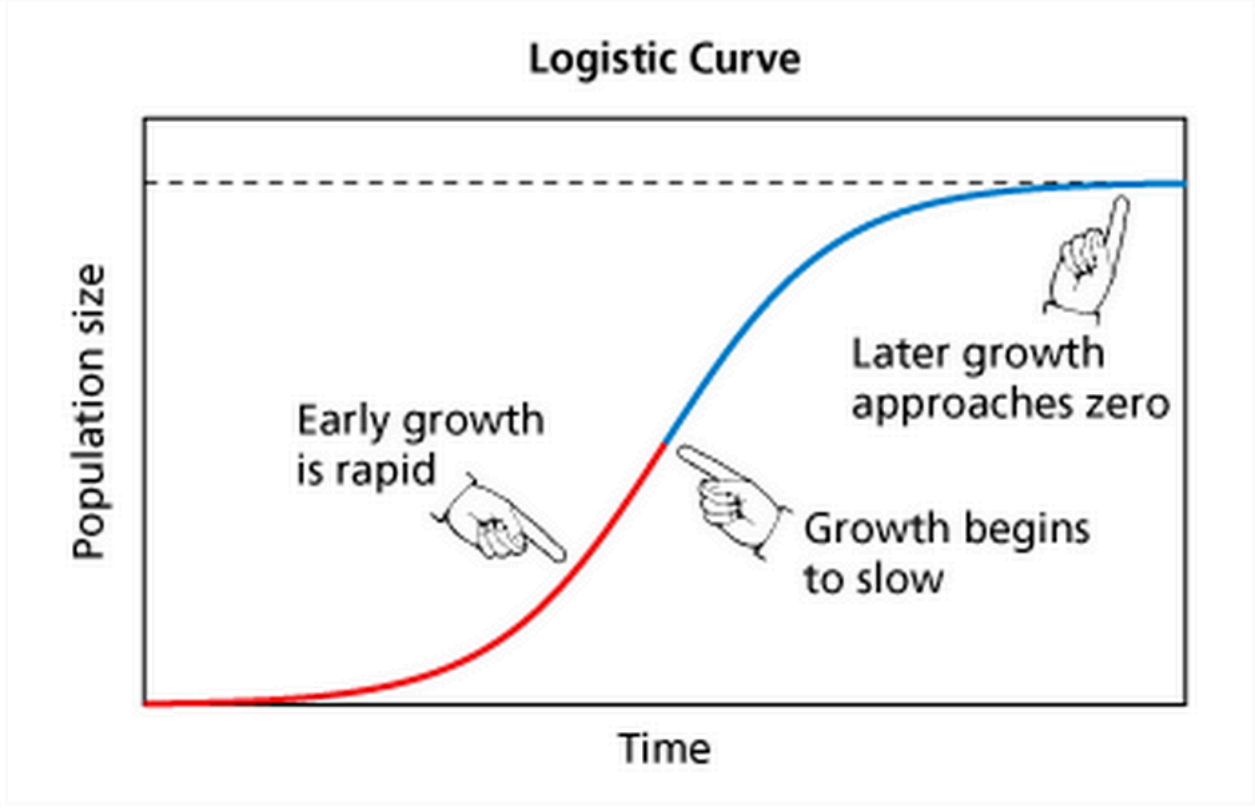
Let's Do Some Calculus

- For a whole-Internet random scanning worm
  - Let v be fraction of addresses that are vulnerable to worm exploit
  - Let S be average scanning rate (syns/hr).
  - Then K = Sv is number of new compromises/hr/ already infected machine.
  - $N = 2^{32}v$ is the total number of machines
  - Fraction of machines comprised at time t is a(t)

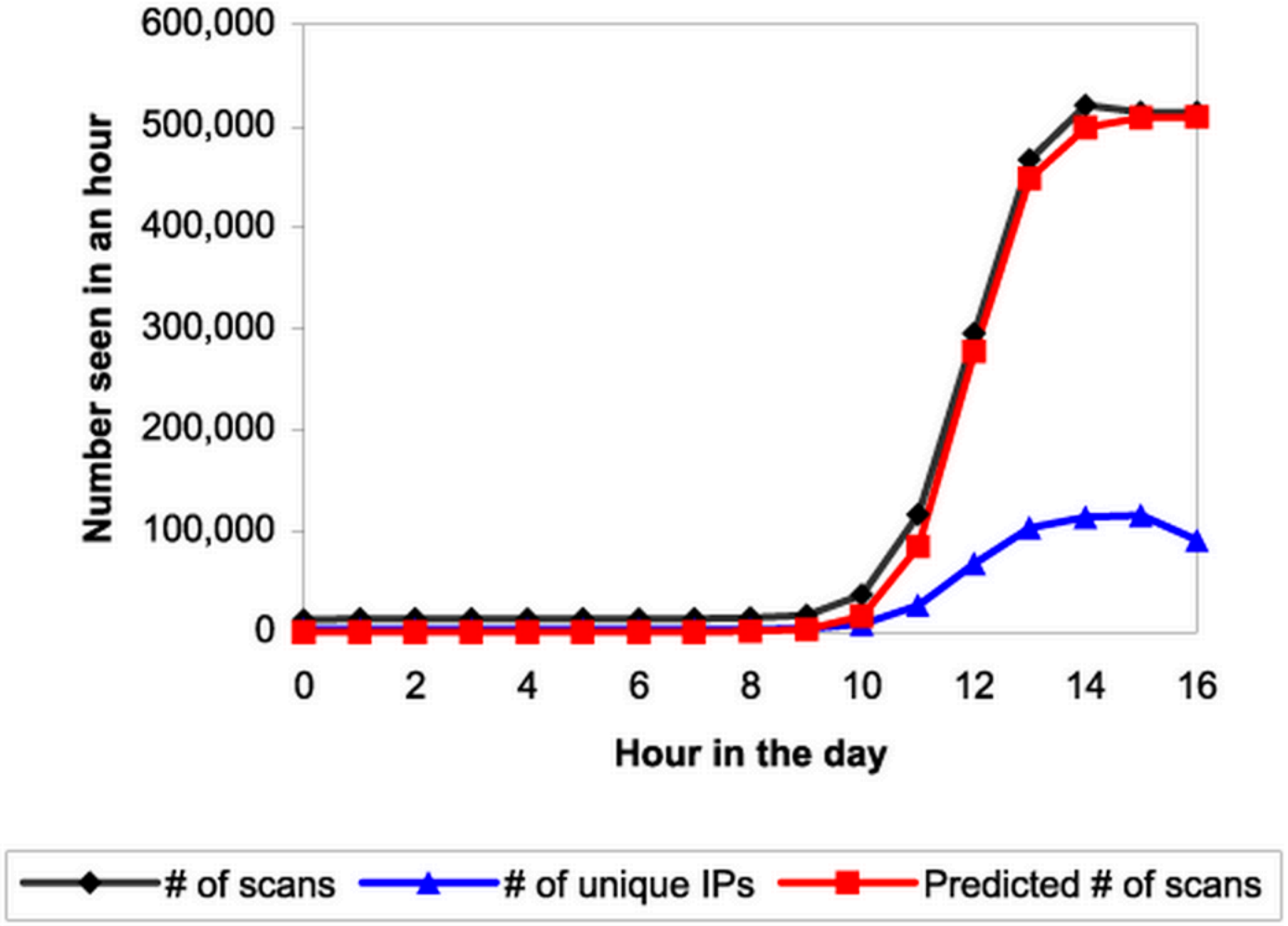
Then…

- How many more machines will be compromised in next dt interval?
  - Nda = NaK(1-a)dt
    - Assumes only infect once ("SI model" of epidemiology)
  - da/dt = Ka(1-a)
  - $a(t) = e^{K(t-T)}/(1+e^{K(t-T)})$
  - Sigmoid or Logistic curve
    - Pierre Verhulst, 19th C.

Logistic Curve



Code Red

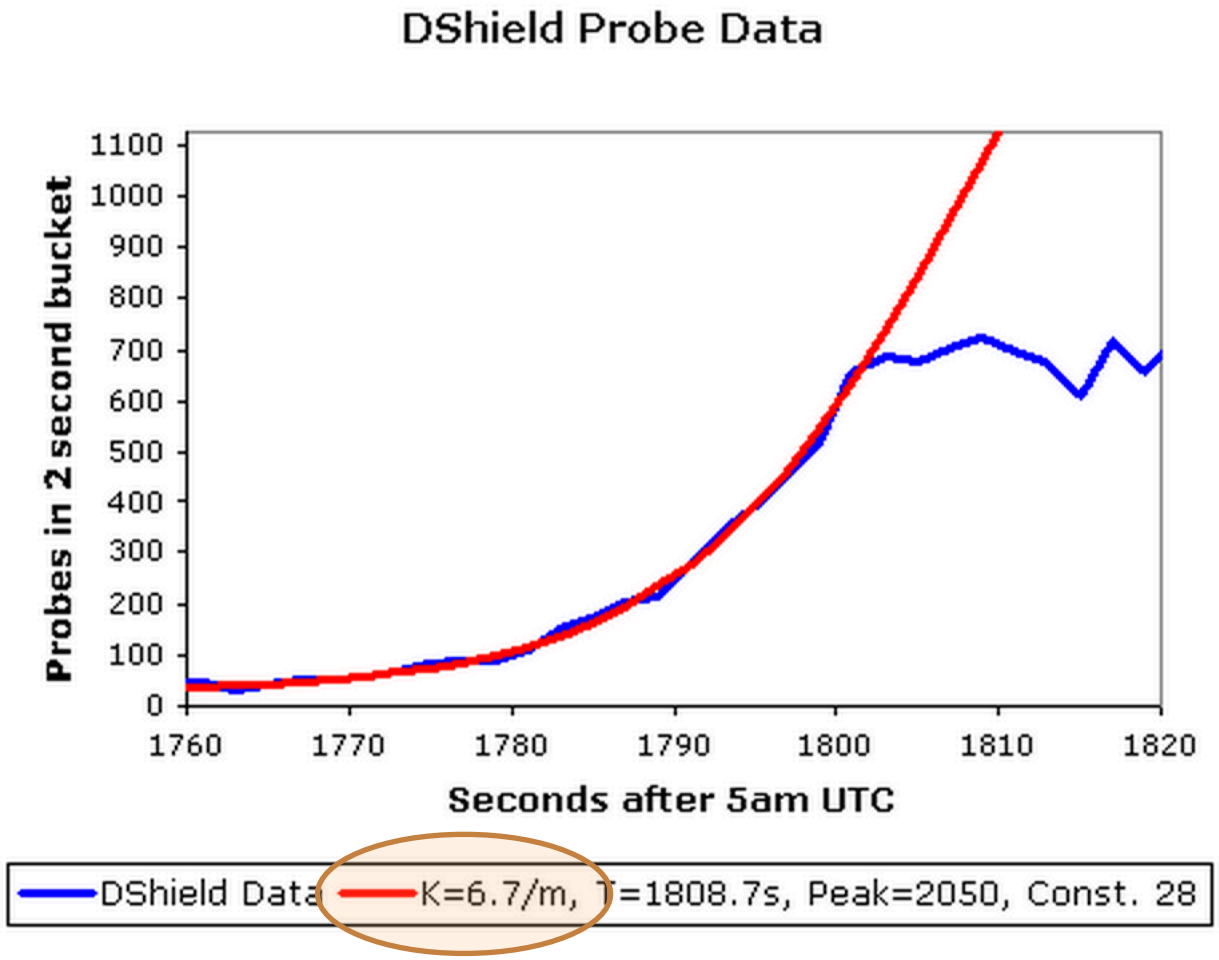


K = 1.8/hr

Code Red Spread

- Vulnerability in IIS Web Server
  - http://www.youtube.com/watch?v=v6GnX3ZhuAg

We thought Code Red was fast, until…



http://www.caida.org/publications/papers/2003/sapphire/sapphire.html

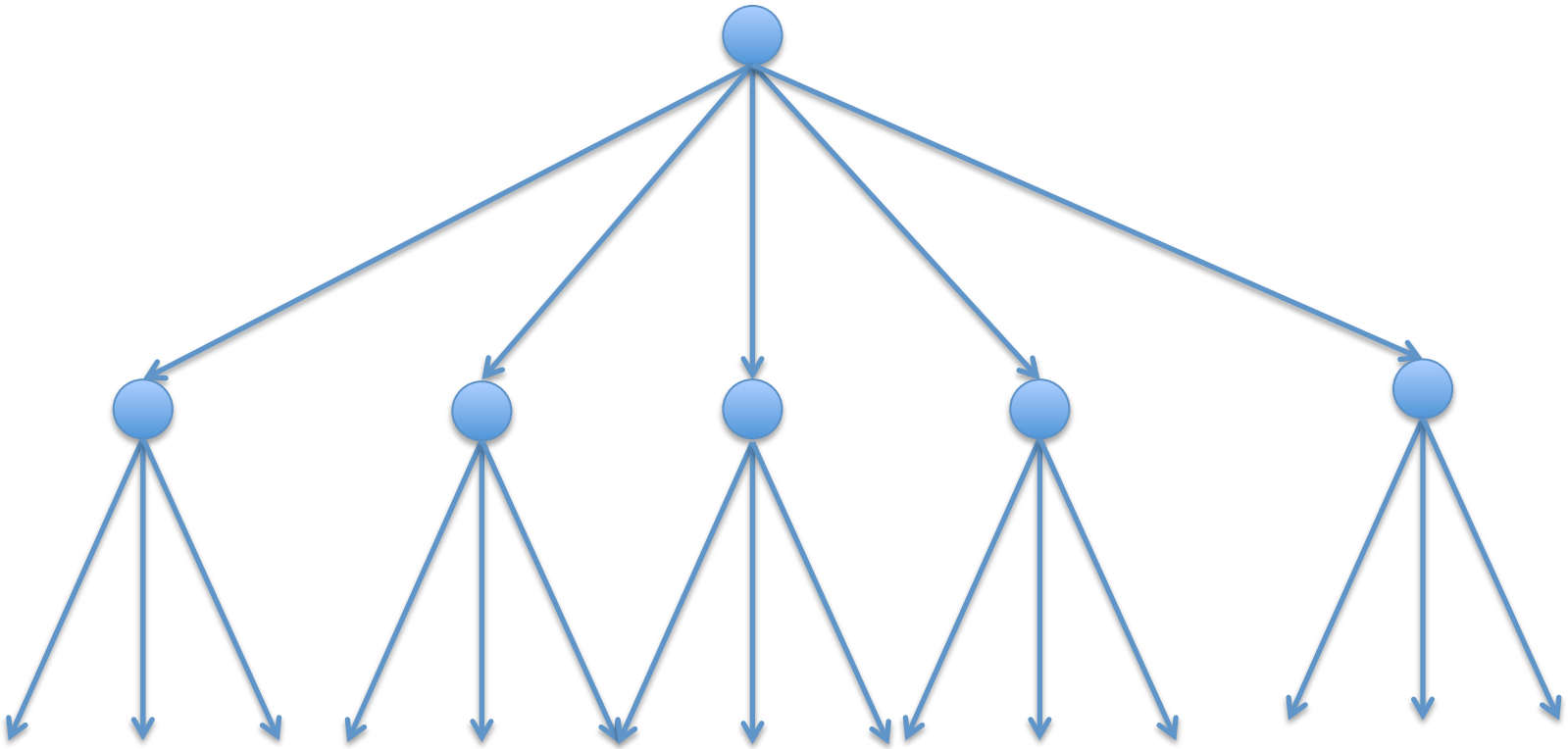
How Slammer Did It

- Occurred in 2003.
- Exploited a vulnerability in MS SQL Server.
- Worm data was only 376 bytes! Fit in one packet.
- Handcrafted machine code contained:
  - Data to overflow buffer and gain control
  - Code to find the addresses of needed functions.
  - Code to initialize a UDP socket
  - Code to seed the pseudo-random number generator
  - Code to generate a random address
  - Code to copy the worm to the address via the socket
- Could spew out hundreds or thousands of worms per second from each infected machine.

How Fast Could a Worm Be?

- Flash worm strategy
  - Do the scanning ahead of time
    - Map entire network/Internet for vulnerabilities
    - (We know intelligence agencies do this)
  - Then precompute spread tree.
    - Each worm instance carries part of address list with it
    - Takes on infecting its part

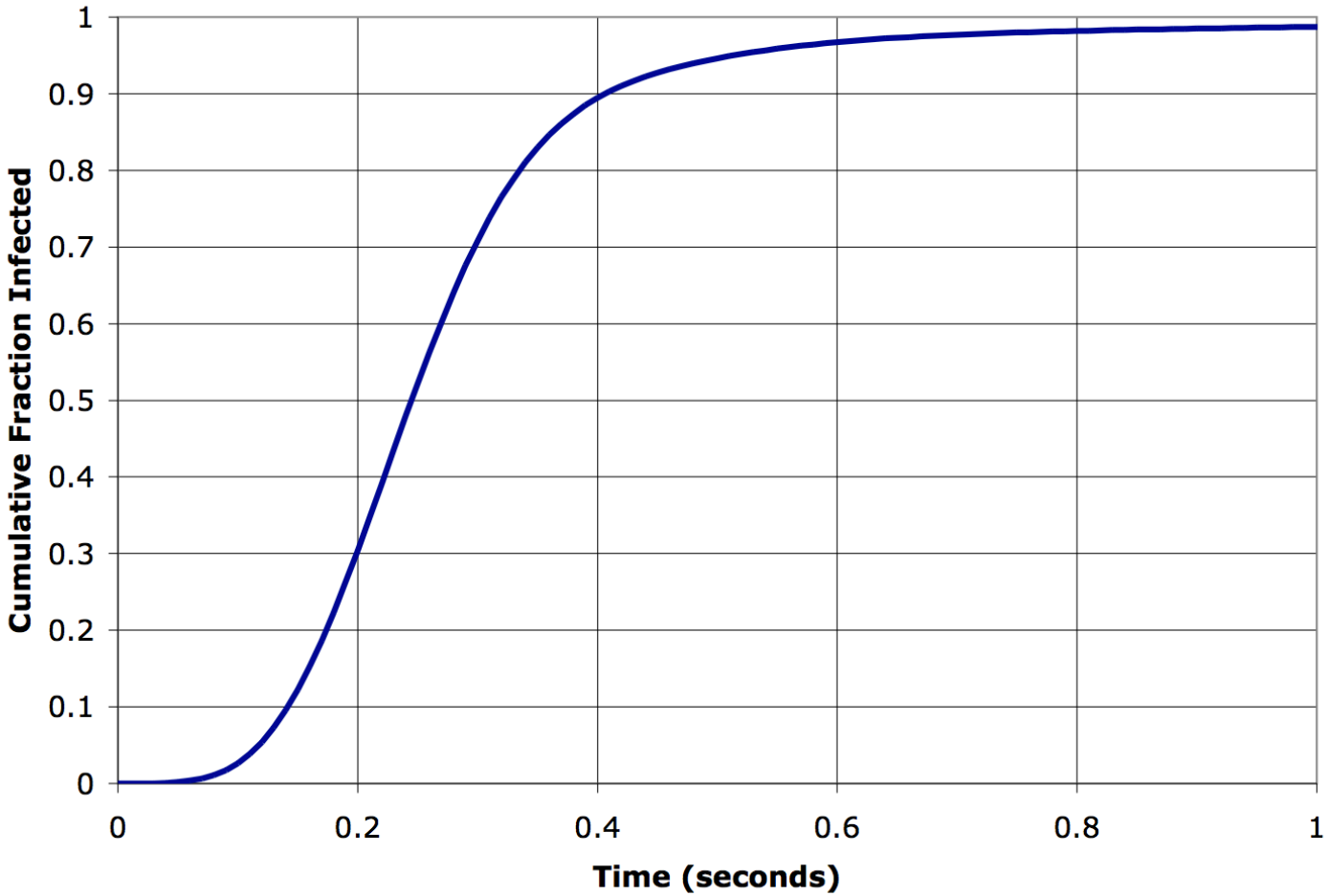
Flash Worm Spread Tree



Strategy

- Pick a very fast node to start on
  - Then a shallow tree
  - Secondary node address lists can fit in one pkt
- Simulation used observed Slammer packet delivery speed distribution
- Observed Internet latency distribution
- Optimum was 9260x107 to infect 1m hosts
- http://www.caida.org/publications/papers/2004/topspeedworms/topspeed-worm04.pdf

Single Packet Flash Simulation



Fraction of a million Internet hosts infected

http://www.caida.org/publications/papers/2004/topspeedworms/topspeed-worm04.pdf

Ultimate constraint

- Speed of light in fiber
  - To go half way around world as crow flies:
    - 6370000*3.14159/3x10^8/(2/3)
    - about 100ms
- Plus time to infect each host

Defenses for Scanning Worms

- Host-level:
  - ASLR/DEP/Canaries/etc
  - Limit outbound connections
- Network level
  - Detect/block scanning
    - Firewalls
    - Packet filtering in routers
    - Intrusion prevention systems
    - In-switch security measures

Overall Dynamic

- Suppose each worm finds r children to infect
  - Total before containment/remediation.
- Successive generations:
  - 1, r, r^2, r^3,…
  - $1 + r + r^2 + r^3 + \ldots = 1/(1-r)$ if $r<1$
  - Eg if r = 0.9, total is 1/(1-0.9) = 10
  - If r > 1, series diverges.
- So must ensure each worm instance finds on average less than 1 child
  - Epidemic peters out
  - Known as “epidemic threshold”
  - Similar to critical mass in nuclear explosions

Overall Dynamic

- Suppose each worm finds r children to infect
  - Total before containment/remediation.
- Successive generations:
  - 1, r, r^2, r^3,…
  - $1 + r + r^2 + r^3 + \ldots = 1/(1-r)$ if $r<1$
  - Eg if r = 0.9, total is 1/(1-0.9) = 10
  - If r > 1, series diverges.
- So must ensure each worm instance finds on average less than 1 child
  - Epidemic peters out
  - Known as “epidemic threshold”
  - Similar to critical mass in nuclear explosions

Email Worms (1999)

- Mostly scourge of late 90s/early 2000s
  - Melissa – Microsoft Word Macro "Worm"
    - Word document attachment to email
    - Used a large variety of enticing subject lines to emails to try to get users to open attachment.
      - Very first version claimed to have passwords to porn sites.
      - Various 'social engineering' hooks to get you to open it
    - Some say not a worm, depending on whether macro language is a "program" or not.
    - Stole address book and mailed itself out

I Love You (2000)

- Subject ILOVEYOU
- Attachment “LOVE-LETTER-FOR-YOU.txt.vbs”
- Scoured address book, so appeared to come from someone you knew.
  - *Many* people opened.
  - Believed to have affected tens of millions of computers.

Email Worms in General

- Are “topological” worms
  - Find their victims using the natural topology of a protocol communication graph
    - In this case email address books
- Use ‘social engineering’
  - Tricking human users into doing something they shouldn’t.
  - In theory could use exploit in mail client, but hasn’t been seen on a large scale.

Email Worm Defenses

- Anti-virus scanning of attachments
- Anti-spam screening of inbound emails
- User education.
  - Including warnings when opening strange attachments.
- Email worms appear not to spread as much any more.
  - Defenses must keep below epidemic threshold.
  - Except…

Storm Worm (2007)

- Used subject lines like
  - "230 dead as storm batters Europe"
  - And many, many others tied to current events
- Had an executable attachment.
  - Defeated AV by "repacking" the exe every 10 minutes.
- Successfully built a large botnet
  - Probably for Russian organized crime.
  - Millions, maybe tens of millions of infected IPs.
- So email worm probably not permanently dead.

Stuxnet Worm

- Discussion today is focussed on spread, not payload.
- Likely target of worm:
  - industrial controllers for centrifuges in Iranian nuclear plant (goal: damage/destroy them).
- Need to
  - Get on internal corporate networks of Iranian entities
  - Get on air-gapped SCADA networks.
  - Find machines attached to right controllers
  - Execute real payload.
- Worm was used as the search strategy to find and cross the bottlenecks.
  - Apparently worked: caused extensive delay to Iranians.

Stuxnet Strategies

- Propagate through any network shares identifiable on accounts of infected computer.
- Zero-day print spooler vulnerability.
- Target hard-coded password in Siemens WinCC (SQL database) product.
- Windows server service vulnerability.
- Ability to infect USB drives.
  - Targetted a Windows vulnerability when viewing the folder on the drive.