Malware: Worms

CS 161 - Computer Security

Profs. Vern Paxson & David Wagner

TAs: John Bethencourt, Erika Chin, Matthew Finifter, Cynthia Sturton, Joel Weinberger

http://inst.eecs.berkeley.edu/~cs161/

April 14, 2010

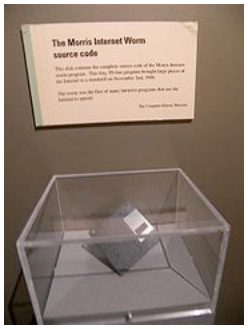
The Problem of Worms

- Virus = code that propagates (replicates) across systems by arranging to be eventually executed
  - Generally infects by altering *stored code*
- Worm = code that self-propagates/replicates across systems by arranging to have itself immediately executed
  - Generally infects by altering or initiating *running code*
  - No user intervention required
- Like with viruses, for worms we can separate out propagation from payload
- Propagation includes notions of *targeting* & *exploit*
  - How does the worm **find** new prospective victims?
  - How does worm get code to **automatically run**?

Studying Worms

- *Internet-scale events*
  - Surprising dynamics / emergent behavior
  - Hard problem of attribution (who launched it)
- Modeling propagation mathematically
- Evolution / ecosystem
  - Shifting perspectives on nature of problem
  - *Remanence*
- "Better" worms
- Thinking about defenses
  - Including "white worms"
- Mostly illustrated from a historical perspective …
  - Details/dates/names for the most part not important
    - Other than **Morris Worm**, **Code Red**, and **Slammer**

The Arrival of Internet Worms



- Internet worms date to Nov 2, 1988 - the *Morris Worm*
  - ***Way*** ahead of its time
- Modern Era begins Jul 13, 2001 with release of initial version of Code Red
- Exploited known buffer overflow in Microsoft IIS Web servers
  - On by default in many systems
  - Vulnerability & fix announced previous month
- Payload #1: web site defacement
  - **`HELLO! Welcome to http://www.worm.com! Hacked By Chinese!`**
  - Only done if language setting = English

Code Red of Jul 13 2001, con't

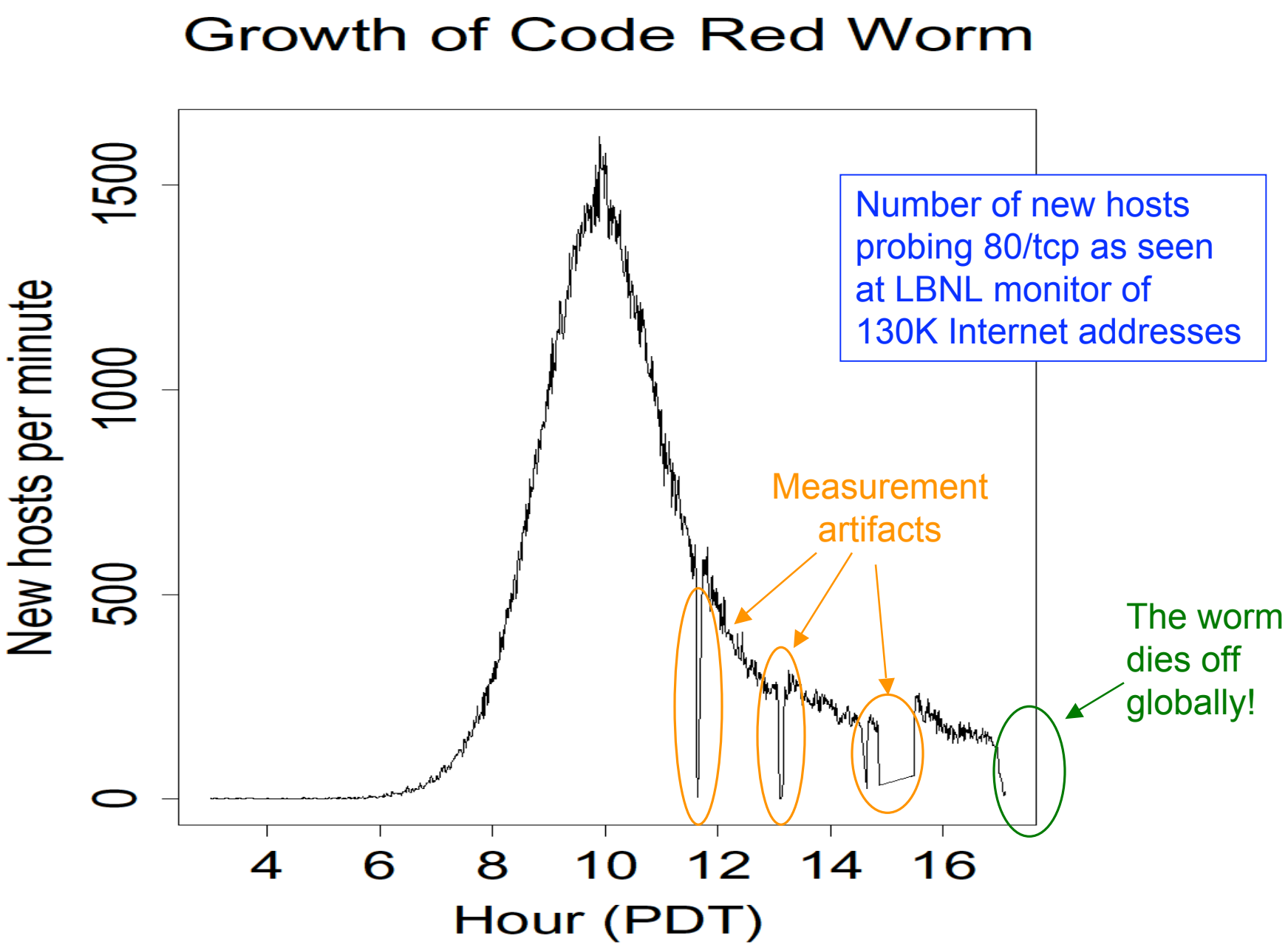
- Payload #2: check day-of-the-month and …
  - … 1st through 20th of each month: spread
  - … 20th through end of each month: attack
    - Flooding attack against 198.137.240.91 …
    - … i.e., *www.whitehouse.gov*
- Spread: via random scanning of 32-bit IP address space
  - Generate pseudo-random 32-bit number; try connecting to it; if successful, try infecting it; repeat
  - Very common (but not fundamental) worm technique
- Each worm uses same random number seed
  - How well does the worm spread?

Linear growth rate

Code Red, con't

- Revision released July 19, 2001.
- White House responds to threat of flooding attack by changing the address of *www.whitehouse.gov*
- Causes Code Red to die for date ≥ 20th of the month due to failure of TCP connection to establish.
  - Author didn't carefully test their code - buggy!

- But: this time random number generator correctly seeded. Bingo!


Growth of Code Red Worm
New hosts per minute
0
500
1000
1500
4
6
8
10
12
14
16
Hour (PDT)
Number of new hosts probing 80/tcp as seen at LBNL monitor of 130K Internet addresses
Measurement artifacts
The worm dies off globally!

Modeling Worm Spread

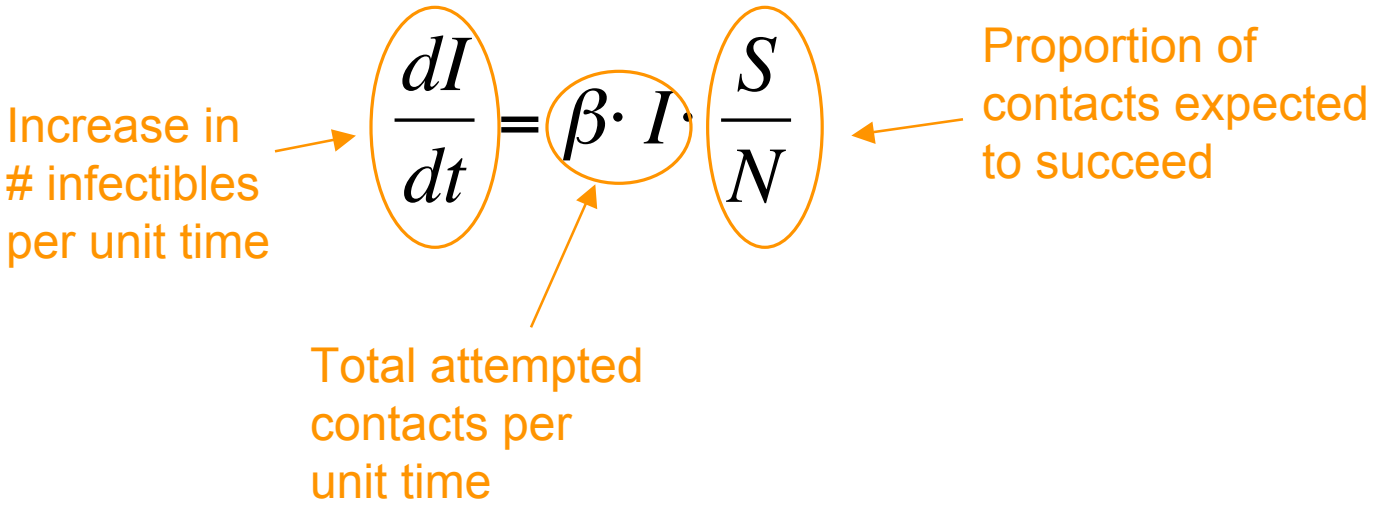
- Worm-spread often well described as *infectious epidemic*
  - Classic SI model: homogeneous random contacts
    - SI = Susceptible-Infectible
- Model parameters:
  - N: population size
  - S(t): susceptible hosts at time t.
  - I(t): infected hosts at time t.
  - β: *contact rate*
    - How many population members each infected host communicates with per unit time
    - E.g., if host scans 10 Internet addresses per unit time, and 2% of Internet addresses run a vulnerable server, then $\beta = 0.2$

$$N = S(t) + I(t)$$
$$S(0) = I(0) = N/2$$

- Auxiliary parameters reflecting the relative proportion of infected/susceptible hosts
  - $s(t) = S(t)/N \quad i(t) = I(t)/N \quad s(t) + i(t) = 1$

Computing How An Epidemic Progresses

- In continuous time:

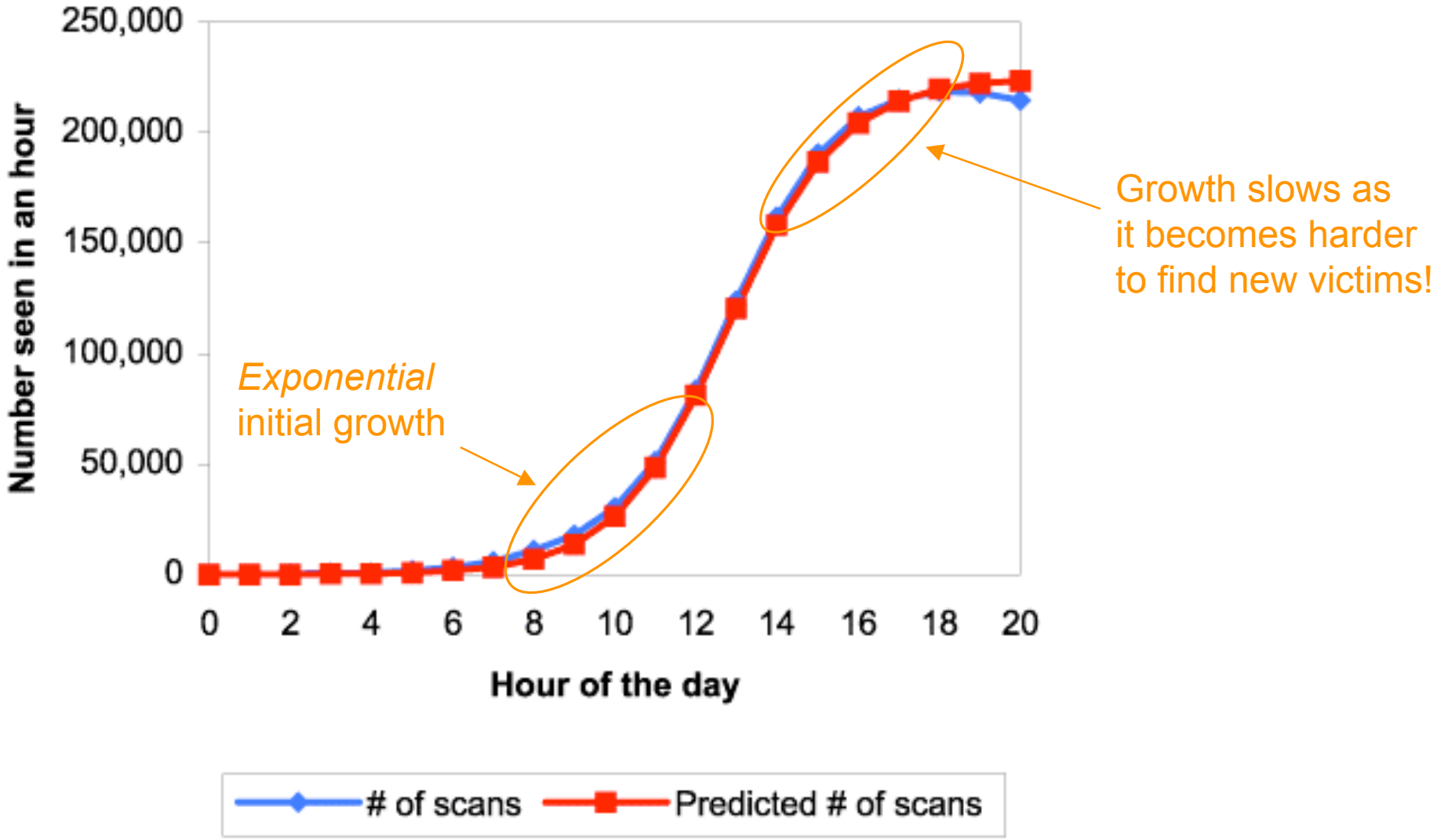


- Rewriting by using i(t) = I(t)/N, S = N - I:

$$\frac{di}{dt} = \beta i(1-i) \quad \Rightarrow \quad i(t) = \frac{e^{\beta t}}{1+e^{\beta t}}$$

Fraction infected grows as a *logistic*

Fitting the Model to Code Red



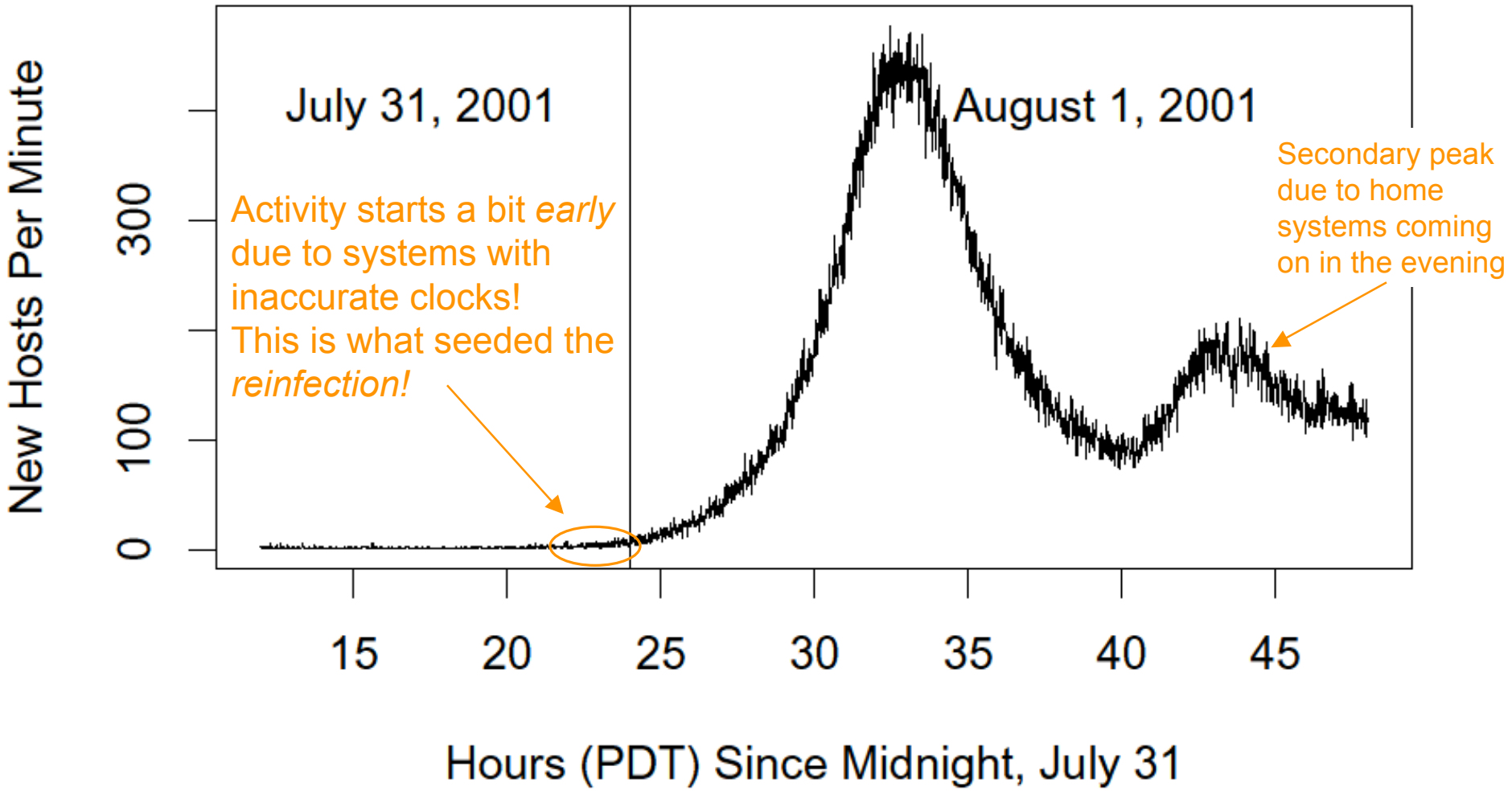
Spread of Code Red, con't

- Recall that # of new infections scales with contact rate β

$$\frac{dI}{dt} = \beta \cdot I \cdot \frac{S}{N}$$

- For a scanning worm, β *increases* with N
  - Larger populations infected more quickly!
    - More likely that a given scan finds a population member
- Large-scale monitoring finds 359,104 systems infected with Code Red on July 19
  - Worm got them in 13 hours
- That night ($\Rightarrow 20^{th}$), worm dies due to DoS bug
- What happens on August 1st?

Return of Code Red Worm



(Again from LBNL monitoring)

Reinfection about 1/2 as big as original

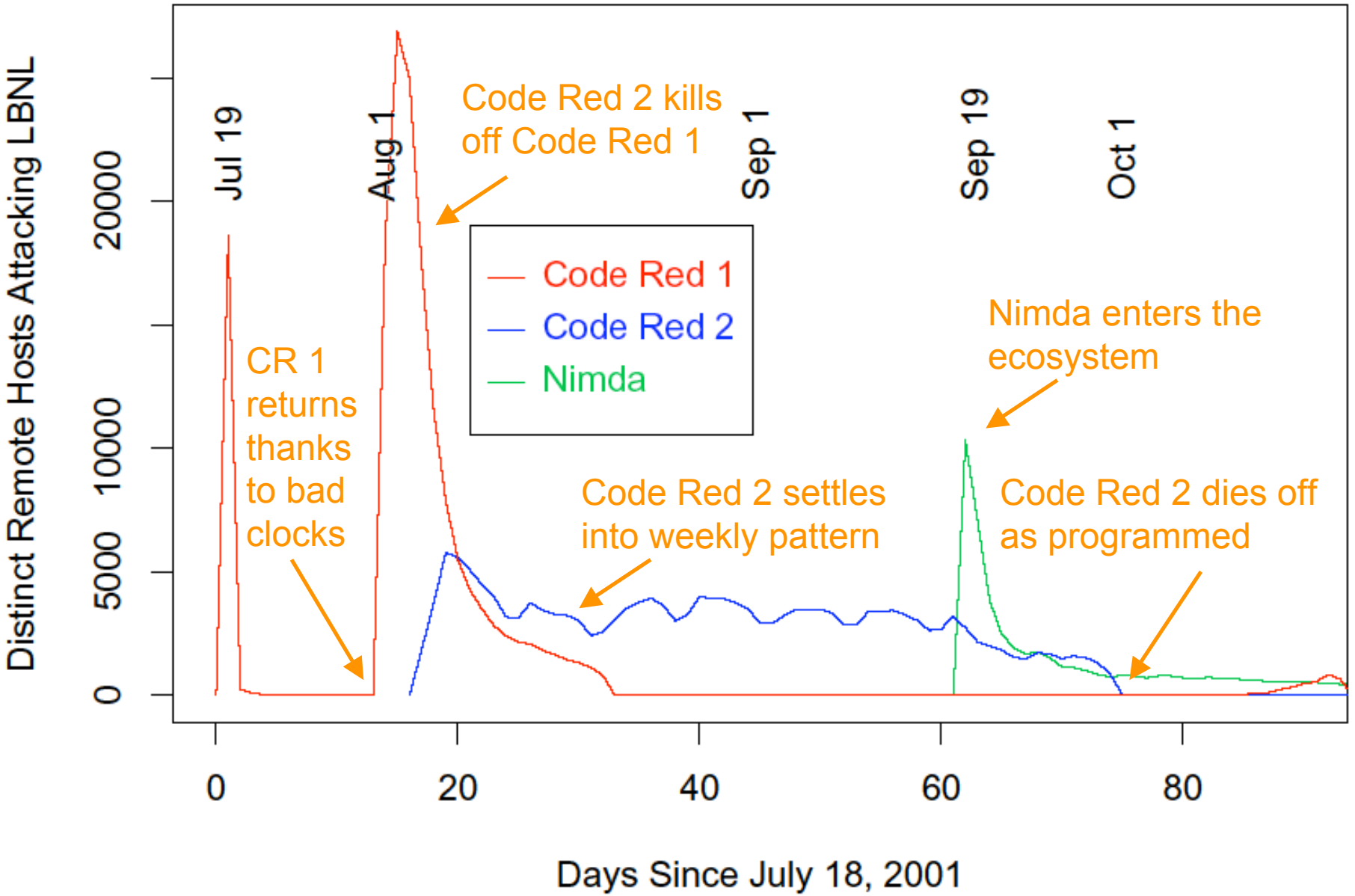
Code Red 2

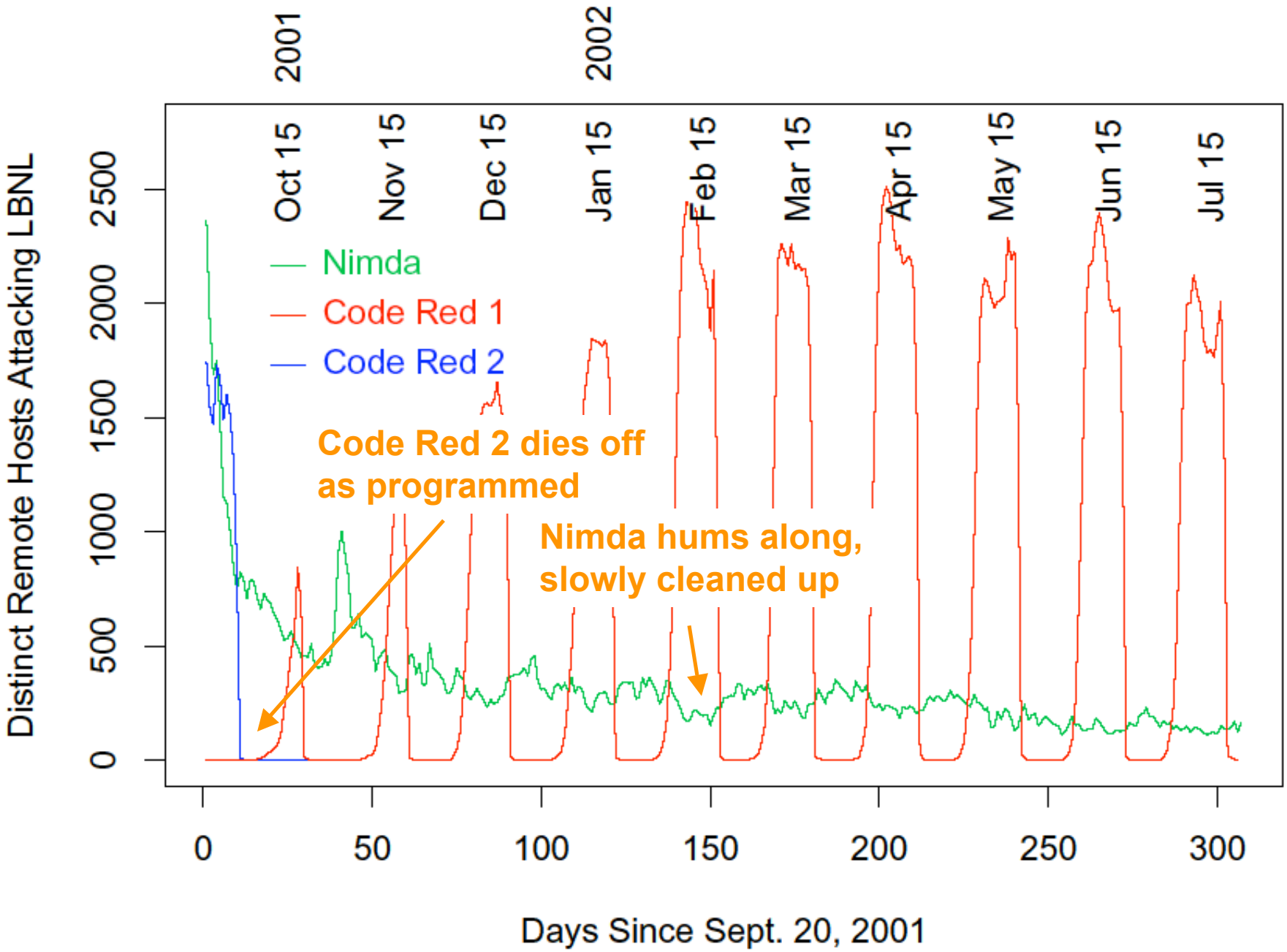
- Released August 4, 2001 (*3 days later!*)
- Exploits same IIS vulnerability
- String inside the code: “Code Red 2”
  - But in fact completely different code base.
- Payload: a root backdoor, resilient to reboots.
- Bug: crashes NT, only works on Win2K.
- Kills original Code Red.
- *Localized scanning*: prefers nearby addresses.
- Safety valve: programmed to die Oct 1, 2001.

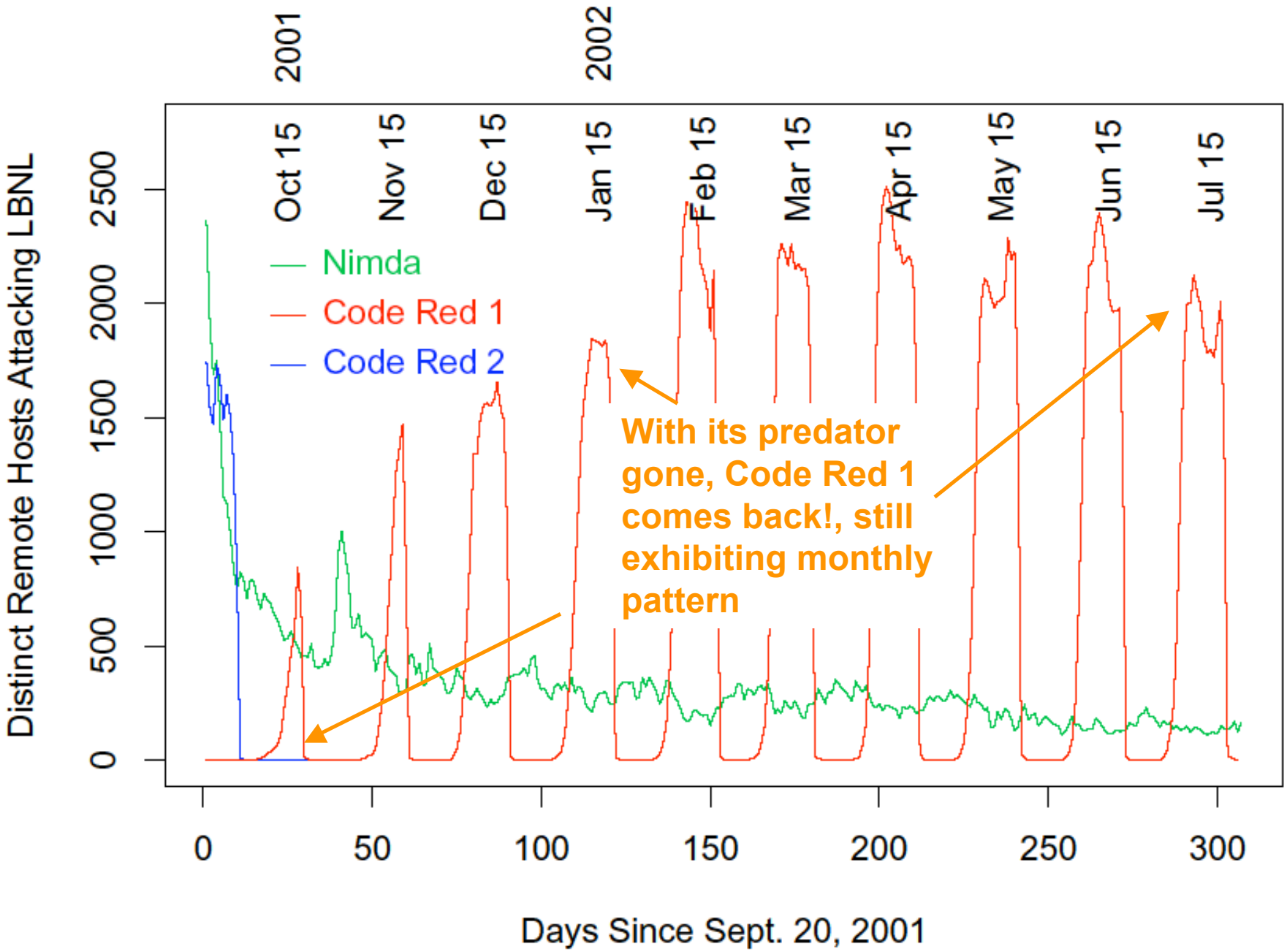
Striving for Greater Virulence: *Nimda*

- Released September, 2001.
- Multi-mode spreading:
  - attack IIS servers like Code Red & Code Red 2
  - email itself to address book as a virus
  - copy itself across open network shares
  - modify Web pages on infected servers with browser exploit
  - scan for Code Red 2 backdoors (!)

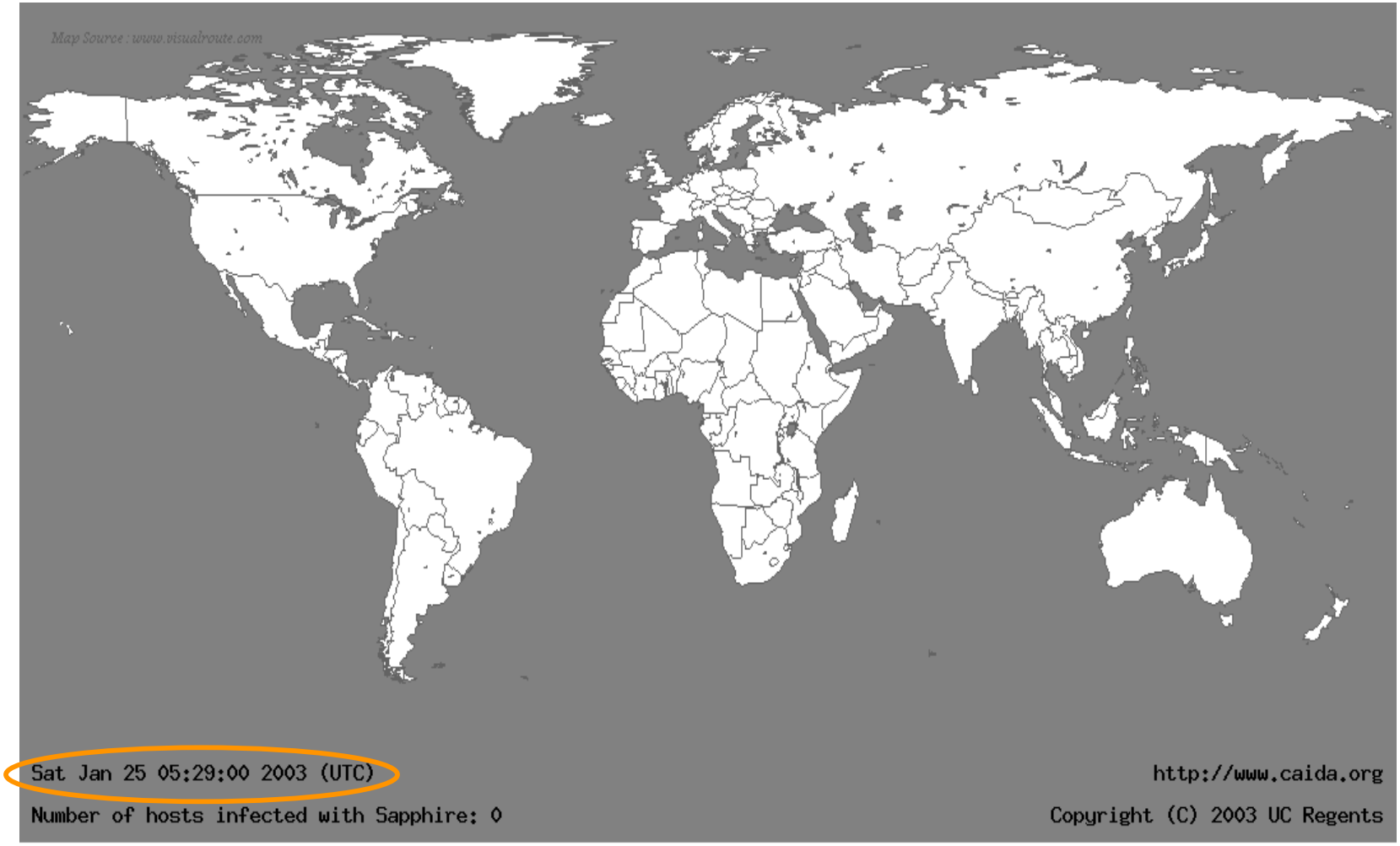
    ⇒ Worms form an *ecosystem*!
- Leaped across firewalls
  - Ravaged sites that lacked “institutional antibodies”


Distinct Remote Hosts Attacking LBNL
0
5000
10000
20000
Jul 19
Aug 1
Sep 1
Sep 19
Oct 1
Code Red 2 kills
off Code Red 1
Code Red 1
Code Red 2
Nimda
CR 1
returns
thanks
to bad
clocks
Code Red 2 settles
into weekly pattern
Nimda enters the
ecosystem
Code Red 2 dies off
as programmed
0
20
40
60
80
Days Since July 18, 2001

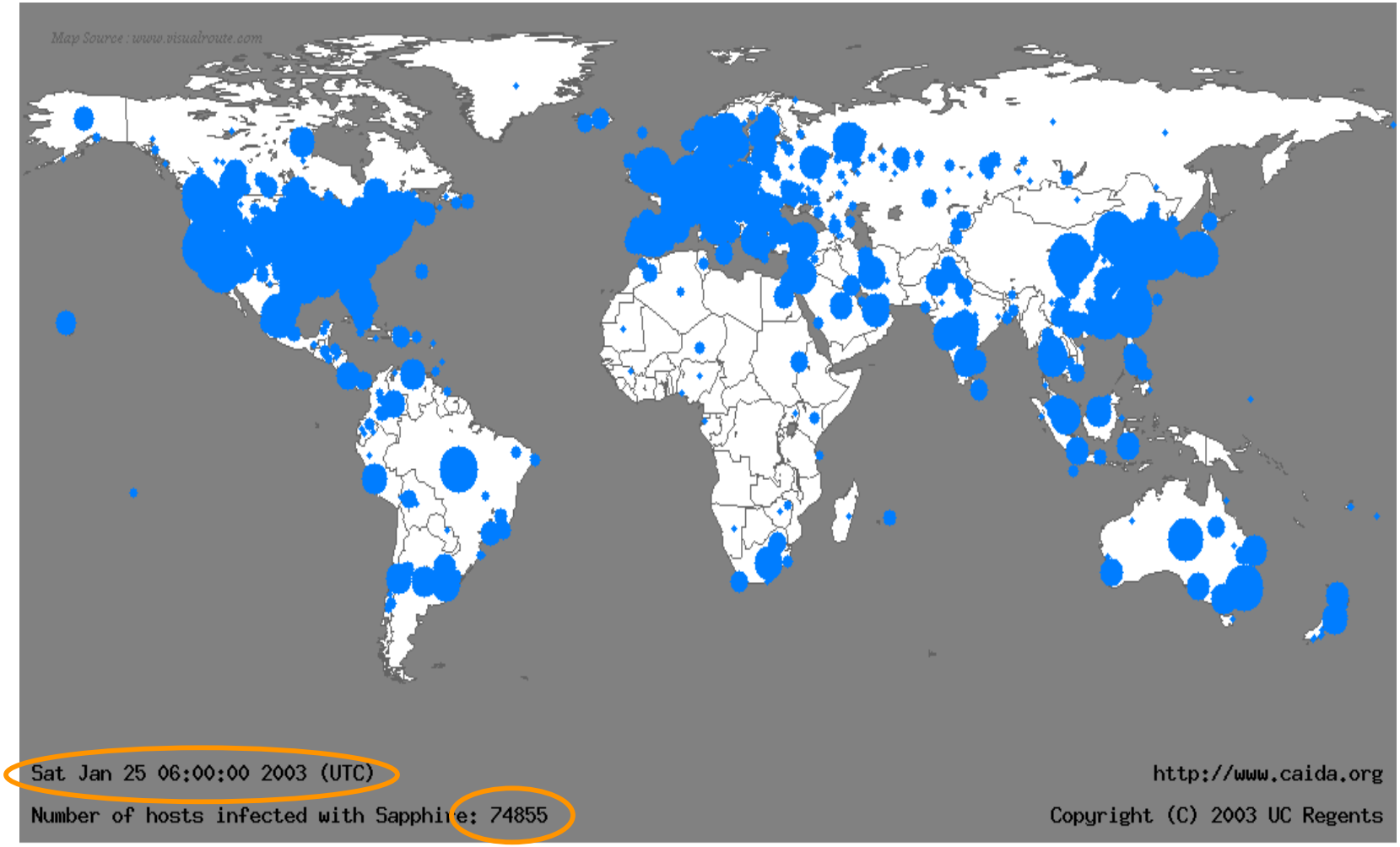

2001
2002
Oct 15
Nov 15
Dec 15
Jan 15
Feb 15
Mar 15
Apr 15
May 15
Jun 15
Jul 15
Nimda
Code Red 1
Code Red 2
Code Red 2 dies off
as programmed
Nimda hums along,
slowly cleaned up
Distinct Remote Hosts Attacking LBNL
0
500
1000
1500
2000
2500
0
50
100
150
200
250
300
Days Since Sept. 20, 2001


2001
2002
Oct 15
Nov 15
Dec 15
Jan 15
Feb 15
Mar 15
Apr 15
May 15
Jun 15
Jul 15
Nimda
Code Red 1
Code Red 2
With its predator gone, Code Red 1 comes back!, still exhibiting monthly pattern
Distinct Remote Hosts Attacking LBNL
0
500
1000
1500
2000
2500
0
50
100
150
200
250
300
Days Since Sept. 20, 2001

Life Just Before Slammer



Life Just After Slammer



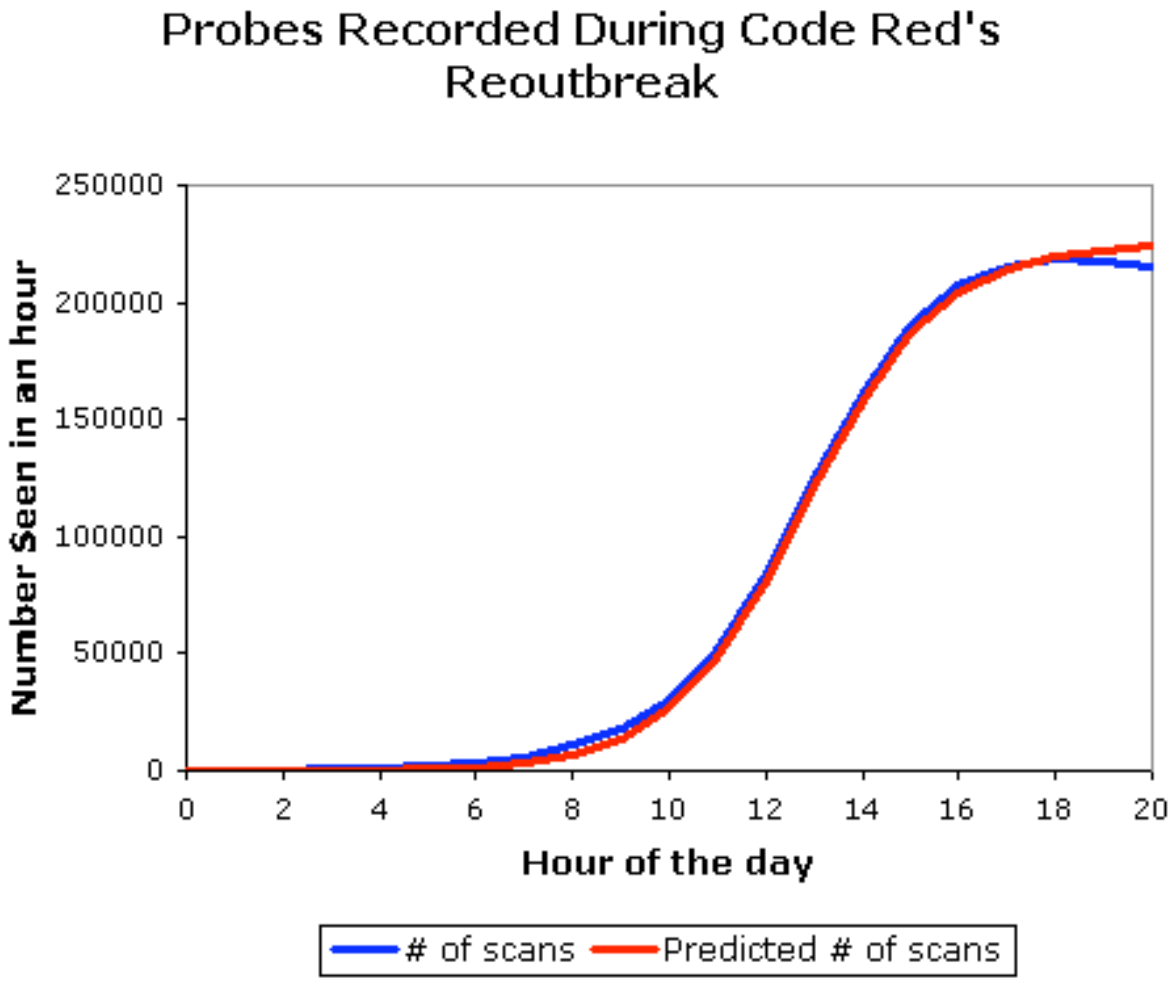
Going Fast: *Slammer*

- Slammer exploited connectionless UDP service, rather than connection-oriented TCP
- *Entire worm* fit in a single packet!

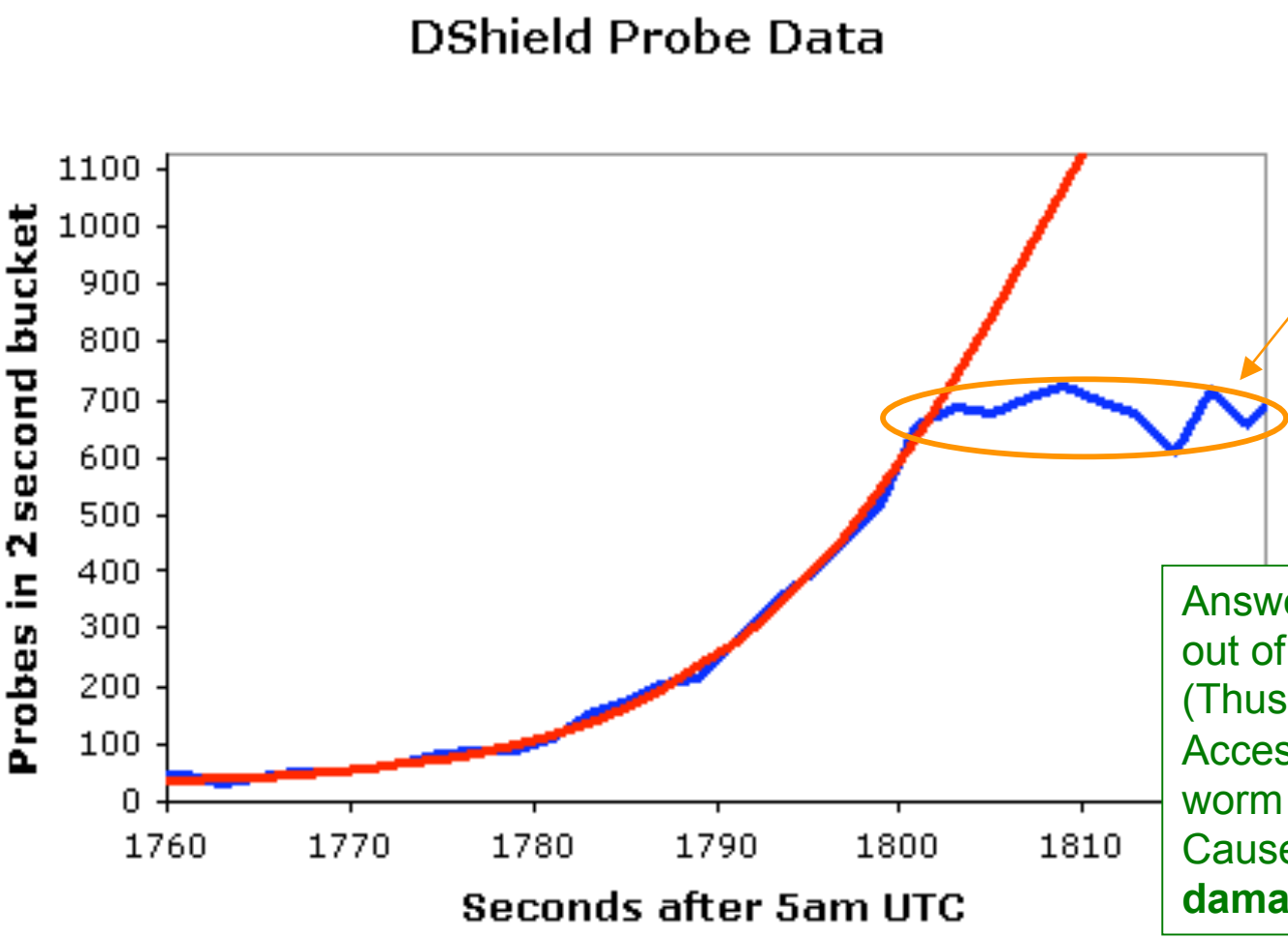
⇒ When scanning, worm could "fire and forget"
Stateless!

- Worm infected 75,000+ hosts in 10 minutes (despite broken random number generator).
- At its peak, **doubled every 8.5 seconds**

The Usual Logistic Growth



Slammer's Growth



What could have caused growth to deviate from the model?

Hint: at this point the worm is generating *55,000,000 scans/sec*

Answer: the Internet ran out of carrying capacity! (Thus, β decreased.) Access links used by worm completely clogged. Caused **major collateral damage**.