

CS574

Computer Security

San Diego State University

Spring 2008

Lecture #8

Today's Structure

- Administrivia
- Questions
- Recent News
- Lecture

Administrivia

- First Exam - February 27
- Assignment #1 due date - March 3

Questions?

Recent News

- Crypto-Gram Newsletter
- Beneficial Worms
- Eradicating Botnets
- Viruses in Digital Photo Frames
- Rogue DNS Servers
- Wireless Security

Lecture

This Lecture

- Worm examples
- More malicious code
- Software flaws
- Controls

Worm Classifications

- Random Scanning Worms
look for someone at random to infect
- Permutation Worms
self-coordinating (divide the IP space)
- Warhol Worms
hit-list scanning - “15 minutes of fame”
- Flash Worms
Internet-sized hit-lists.

Worms - Examples

- “Morris Worm” - 1988 - ARPANET/Various
- “Code Red” - 2001 - Internet/IIS
- “Nimda” - 2001 - Internet/Various
- “Sapphire” - 2003 - Internet/MS-SQL
- “Blaster” - 2003 - Internet/MS-RPC
- “Witty” - 2004 - Internet/Blacklce
- “Storm” - 2007 - Internet/Various

The Morris Worm

Morris Worm

- November 2, 1988
- ~6,000 hosts on the Internet/ARPANET
- BSD UNIX and derivative Systems
- Attacked SUNs and VAXes only

Morris Worm Vectors

- sendmail via debug mode
- fingerd (on VAXes) via buffer overflow
- trusted machines via rexec and rsh
- guessing passwords via rexec and rsh

Morris Worm Attack

- bootstrap by getting target system to run a script
- compile the “grappling hook”
- fetch full worm from the infecting machine
- camouflage
 - rename program and arguments
 - delete disk images and zero program memory
 - only execute a short while, then fork()
- start attacking other machines

Morris Worm Controls

- recompile sendmail without DEBUG
- shut down or patch fingerd
- move/rename compiler and linker programs
- enforce password policies and/or disable the “r-services”

Morris Worm

Effects

- Many sites “took themselves off the ‘Net’”
- Many other sites just shut down mail
- Excessive network traffic
- “Accidental” resource exhaustion

Morris Worm Lessons

- Connectivity was important to learning about the worm and how to fix infected or vulnerable machines.
- People called or sent email to those they knew and trusted – the “old boy network”.
- Who do you call late at night? How do you authenticate yourself?
- Speakerphones, conference calls, and rumor control were important: tools were less important than people.

Morris Worm

Lessons

- Author of the worm was an insider in the computer security field – restricting information would not have helped.
- Diversity was good. *Some* systems stayed up and allowed for communication.
- Backups can be an easy way to recover.
- Logging is important.

Morris Worm Lessons

- Denial of Service attacks are easy.
- It seems like a good idea to have a central location for security warnings and fixes.
- Managing the Press is important (so those who are busy fixing the problem aren't continually being interrupted for statements).
- Knee-jerk reactions should be avoided.

The Code Red Worm

Code Red Worm

- July 2001
- Attacked MS-Windows machines running the IIS web-server.
- At least three variants.
- Very fast – ~350K machines in ~14 hours.
- Infected base doubled every ~37 minutes.

Code Red Worm Vector

- Used a vulnerability in ISAPI library used by IIS.
- HTTP GET request caused buffer overflow.
- Time-sensitive payload:
 - day 1-19 - infect phase
 - day 20-27 - attack phase
 - remainder - dormant

Code Red Worm Payload

- Code Red I
 - defaced web-pages (sometimes)
 - attack phase packet-flooded an IP address
- Code Red II
 - installed a back door only
 - no defacement
 - no denial-of-service

The Nimda Worm

Nimda Worm

- Also in 2001
- Microsoft Windows systems only.
- Multi-vector worm (email, directory access, IIS)
- Variable subject lines and MD5 checksum.
- Very messy (best fix was to reinstall from known-clean media)

Nimba Worm Vectors

- Email attachments
- Open network shares
- Web-server to client
- Client to web-server using 'directory-traversal' vulnerabilities
- Client to web-server using Code Red II backdoors.

Nimbda Worm

Payload

- Harvest email addresses for email attack.
 - subject lines varied
 - only sent out attack once every 10 days
- Scan for vulnerable IIS servers
 - 50% – only vary last to octets
 - 25% – only vary last three octets
 - 25% – choose totally random IP address
- Infect local binaries (programs)

Nimda Worm Payload

- Walk the directory structure, creating copies of the worm as “readme.eml” or “readme.nws”
- When “web content” identified, add Javascript fragment to download and run the worm.
- Share the C: drive.
- Create a “Guest” account and add it to the Administrator Group.

The Sapphire Worm

Sapphire Worm

- January 2003
- Also called “Slammer”
- Attacked MS SQL Server
- *Very small* – 376 bytes – fits in one UDP packet.
- *Very fast* – infected based doubled every 8.5 seconds.
- 90% of vulnerable hosts infected in first 10 minutes.

Sapphire/Slammer Vector

- 376-byte payload to port 1434/udp.
- Stack buffer overflow.
- Random scanning (choose IP at random)

Sapphire/Slammer Payload

- Nothing explicitly malicious.
- Detrimental effects due to overloading of networks and taking databases out of operation.
- 55 million scans in just 3 minutes.

The Blaster Worm

Blaster Worm

- Also in 2003
- MS Windows RPC/DOM
- DDoS against windowsupdate.com

Blaster Worm

Vector & Payload

- Buffer overflow against Microsoft's RPC (Remote Procedure Call) service.
- Obtains a copy of "msblast.exe" from the infecting host and runs it.
- Scans for other vulnerable hosts.
- Date-activated DDoS attack (avoided with DNS tricks) against source of patches.

The Witty Worm

Witty Worm

- March 2004
- Internet Security Systems “BlackIce” and “RealSecure” intrusion-protection products.
- General MSWindows users not endangered, only those who were proactively trying to defend themselves with these products.
- Alert on March 18, attack started March 19.

Witty Worm Vector & Payload

- Buffer-overflow in ICQ-parsing code.
- Sends 20K varying-sized packets to random IP address. (Worm size is 637 bytes.)
- Writes 0xFFFF to random locations on disk.

Witty Worm Propagation

- Peak infection within 45 minutes.
- 90Gbits/second of traffic a peak.
- Relatively small set of vulnerable hosts.
- Apparently seeded with ~110 initial hosts.
- Self-limiting (due to destructive payload).

The Storm Worm

Storm Worm

- January 2007
- MSWindows machines targeted
- For profit, not fame
- 1 to 5 million infected machines
- Partitioning of the botnet?

Storm Worm Vector & Payload

- Email, web, and hijacked chat sessions.
- Uses variable (and plausible) subject lines.
- Modifies existing and installs new drivers.
- Coordinated attack in other machines.
- Compromised machines primarily used for sending spam and DDoS.

Storm Worm Features

- Under continual development.
- Command and control system is sophisticated, obfuscated, and P2P.
- Aggressive automatic defense.
- Infected instances remotely upgraded.
- Encrypted executables.

Lecture References

- Pfleeger, *Security in Computing*, 3rd & 4th edition
- Bishop, *Introduction to Computer Security*
- *With Microscope and Tweezers: An Analysis of the Internet Virus of November 1988*
<http://www.mit.edu/people/eichin/virus/strategies.html> (available as of february 2008)
- Porras, et. al, *A multi-perspective Analysis of the Storm (Peacomm) worm*, SRI Oct 10 2007

End Lecture

Reading

(For the next few classes)

- Pfleeger, Chapter 3 (finish)
- Pfleeger, Chapter 4 (start)
- <http://www.enderunix.org/documents/eng/bof-eng.txt>
- <http://www.phrack.org/archives/49/P49-14>

Finis