

# Malicious Code

CSH5 Chapter 16  
"Malicious Code"

Robert Guess & Eric Salvaggio

## Topics

- Introduction
- Malicious Code Threat Model
- Survey of Malicious Code
- Prevention of Malicious Code Attacks



CSH5 Chapter 16: "Malicious Code"

## Introduction

- Malicious code / logic
  - ❑ *Malware*
  - ❑ Hardware, software or firmware intentionally included or inserted in system for unauthorized purpose
- Classification may be difficult
  - ❑ Categories overlap because malware may have multiple functions and attributes
    - ✓ E.g., virus / worm / Trojan horse / spyware
- Some code may not be intended as malware by creators
  - ❑ Context and intent determine whether code is viewed as malicious



## Malicious Code Threat Model (1)



- Actor: structured or unstructured threats
  - ❑ Individuals, organizations, nation-states
- Access: allowed physical or logical path
- Asset: resource of interest
- Action: execution of malicious code or logic
- Outcome
  - ❑ Intelligence, surveillance, reconnaissance
  - ❑ Disruption of operations
  - ❑ Destruction of assets
  - ❑ Publicity for cause
  - ❑ Negative publicity against victim



## Malicious Code Threat Model (2)

- Self-replicating Code
- Actors: Origin of Malicious Code Threats
- Actors: Structured Threats
- Actors: Unstructured Threats
- Access vs Action: Vector vs Payload



## Self-replicating Code

- Not inherently malicious
- Early experiments (1960s) had no evil intent
  - ❑ Darwin (1961) involved memory worms
  - ❑ Self-replicating code
  - ❑ Competition → resource exhaustion



## Is a Beneficial Virus Possible?



- Ideas for beneficial self-propagating code:
  - ❑ Distribute antivirus programs automatically through Internet
  - ❑ Install patches on servers in networks
  - ❑ Distribute useful information automatically
- Problems:
  - ❑ What if there's a bug or incompatibility in the self-propagating code?
  - ❑ What if the patches are not appropriate for a specific server or network?
  - ❑ What if the recipient does not see the info as useful?



Copyright © 2011 M. E. Kabay. All rights reserved.

7

## Is Writing Malware Illegal?



- No explicit law against *writing* malicious code
- No illegality even in *sharing* such code among willing recipients
- Current efforts to define statutes
  - ❑ Based on laws banning possession of burglary tools (e.g., lock picks)
  - ❑ Require registration and licensing of locksmiths
  - ❑ Would treat malware and Trojans in same way
  - ❑ No significant progress to date



Copyright © 2011 M. E. Kabay. All rights reserved.

8

## Actors: Origin of Malicious Code Threats



- Structured threats
  - ❑ Nation-states
  - ❑ Corporate criminals
  - ❑ Organized crime
- Unstructured threats
  - ❑ Rogue actors; e.g.,
    - ✓ Individuals
    - ✓ Script kiddies



Copyright © 2011 M. E. Kabay. All rights reserved.

9

## Actors: Structured Threats



- Well-funded, systematic
- Industrial espionage, information operations, large-scale fraud & theft
- Organized crime responsible for 90% malware
  - ❑ Extortionists target online gambling
  - ❑ Pump 'n' dump schemes cost \$B
  - ❑ Industrial espionage using spyware growing
- China major player
  - ❑ Major source of attacks
  - ❑ PRC PLA doctrine emphasizes asymmetric warfare using information technology
  - ❑ Total government control over hacking



Copyright © 2011 M. E. Kabay. All rights reserved.

10

## Actors: Unstructured Threats



- Random
- Relatively limited
- Does not target national security
- Relatively minor



Copyright © 2011 M. E. Kabay. All rights reserved.

11

## Access vs Action: Vector vs Payload



- **Vector**
  - ❑ Agent is avenue of access
  - ❑ Physical access via people who can enter premises
  - ❑ Network access via Web server, client systems, e-mail attachment, portable device
- **Payload**
  - ❑ Function (action) inserted in system
  - ❑ Malicious logic, remote access software, remote control software



Copyright © 2011 M. E. Kabay. All rights reserved.

12

## Survey of Malicious Code

- Viruses
- Worms
- Trojans
- Spyware & Adware
- Rootkits
- Bots & Botnets
- Malicious Mobile Code



13

Copyright © 2011 M. E. Kabay. All rights reserved.

## Virus Mechanisms

- Boot sector: sector 0 of disk
- File infector: inserts JUMP instruction, adds code, returns to original location and continues loading
- Macro virus: exploits weakness of MS scripting language in Word, PowerPoint, Excel, Access etc.

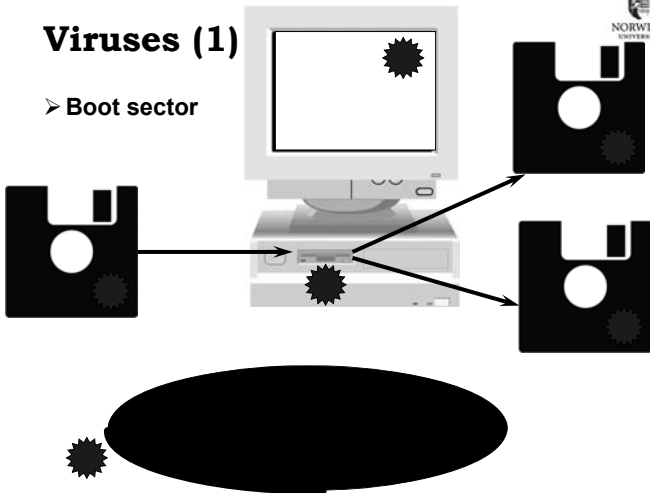


14

Copyright © 2011 M. E. Kabay. All rights reserved.

## Viruses (1)

- Boot sector

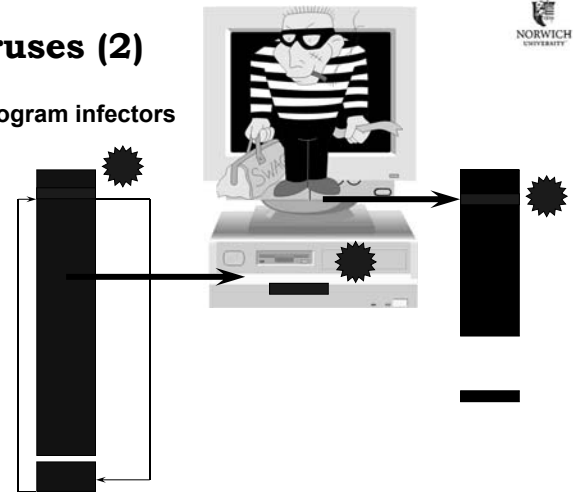


15

Copyright © 2011 M. E. Kabay. All rights reserved.

## Viruses (2)

- Program infectors

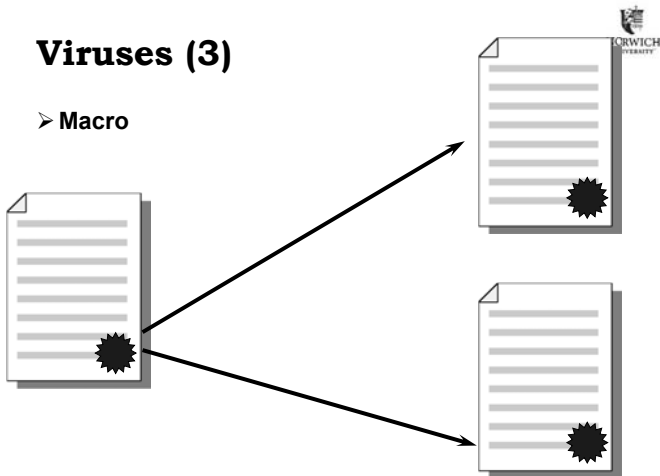


16

Copyright © 2011 M. E. Kabay. All rights reserved.

## Viruses (3)

- Macro



17

Copyright © 2011 M. E. Kabay. All rights reserved.

## 1995-1996: Early Macro Viruses

- MS-Word macro virus (concept) released Aug 95
  - ❑ MS-Word macro viruses now (2009) more than half all infections in the wild
  - ❑ About 150 types known to date
- MS-Excel virus discovered June 96
  - ❑ Anti-virus available within days
  - ❑ Spreading more slowly than Word macro viruses because of lower rate of exchange of spreadsheets

18

Copyright © 2011 M. E. Kabay. All rights reserved.

## 1999-03: Melissa Virus

- Friday 26 March: CERT-CC initial reports of fast-spreading new MS-Word macro virus
- Melissa written to infect Word documents
  - Uses victim's MAPI-standard e-mail address book
  - Sends copies of itself to first 50 people on list
  - E-mail message w/ subject line "Important Message From <name>"
  - Spread faster than any previous virus
  - Followed by similar e-mail-enabled viruses

## Viruses (4)



- Logic Bombs
  - ❑ Any malicious code, replicating or not, that delivers a payload as a result of a logic test (e.g., specific date, absence of employee record)
  - ❑ Time bombs (set off on a date or category of date) are a subset of logic bombs
- Cross-site scripting malware
  - ❑ Exploit flaws in Web application servers & client code
  - ❑ In 2005, "Samy" created script that generated >1M "friends" on MySpace using flaw in Internet Explorer to use JavaScript insertion exploit
  - ❑ Sentenced to 3 years probation, 90 days community service

## Viruses (5)

- Polymorphic viruses
  - ❑ Intended to defeat signature-based antivirus tools
  - ❑ Modify themselves at time of replication
- Polymorphic Engine
  - ❑ Encrypts code
  - ❑ Includes self-decryption capability
  - ❑ Dark Avenger wrote MtE (aka Mutation Engine) in late 1980s
    - ✓ Programmer from Sofia, Bulgaria
    - ✓ Detested Vesselin Bontchev, famous AV expert
    - ✓ Also attacked researcher Sarah Gordon by name



## Viruses vs Worms

- Viruses integrate into host code
  - ❑ Replicate upon execution of infected code
- Worms are free-standing code
  - ❑ Replicate via networks
  - ❑ E-mail (e.g., Outlook) especially common vector
- Some worms have viral properties
  - ❑ Integrate themselves into e-mail messages and convert them to executable files
  - ❑ Frequently conceal executable file type
  - ❑ Depend on default suppression of file suffix (e.g., *AnnaKournikova.jpg.vbs.txt*)



## 1987: IBM Christmas Tree Worm

- E-mail sent via IBM internal e-mail network
- Included program to draw ASCII Christmas tree on screen
- Used recipient's e-mail address book to mail itself to everyone on the network
- No mechanism to prevent superinfection
- Overloaded worldwide IBM networks
- Messages escaped from IBM into BITNET



## 1988: The Morris Worm

- Robert T. Morris (not a "Robert T. Morris, Jr"!)
- ❑ Cornell University grad student
- ❑ Son of famed NSA cryptographer Robert H. Morris
- ❑ Wrote paper on *sendmail* and *fingerd* vulnerabilities on UNIX systems
- ❑ Seems to have intended to demonstrate significance
- Released a defective version of his demo worm
  - ❑ Originally intended to replicate slowly, avoid superinfection
  - ❑ In fact grew fast and superinfected systems worldwide



## Morris Worm (cont'd)

- Launched Worm at 17:00 on 2 November 1988
- By 06:00 next morning the Internet was effectively down
  - ❑ ~6,000-9,000 systems crashed or taken offline
- Computer scientists worked feverishly all night analyzing the Worm
  - ❑ Distributed fixes by telephone and fax (no 'Net)
  - ❑ Led to formation of CERT-CC® in Dec 1988
- Morris convicted of violating 1986 Computer Fraud and Abuse Act (18 USC §1030)
  - ❑ 400 hours community service + \$10K fine

## 1999-12 W.95.Babylonia Virus/Worm

- Extensible virus
- Payload modified remotely
- Trojan virus-dropper
  - ❑ Disguised as Y2K bug fix for internet relay chat (IRC) users
- Sent itself other users
- Polled Internet site in Japan
  - ❑ Looked for updated plugins

## 2000-05: ILOVEYOU Worm



- E-mail subject *ILOVEYOU*
- E-mail attachment *LOVE-LETTER-FOR-YOU.TEXT.vbs*
- Used all addresses in address book
- Became #1 infectious code in Europe, Asia, USA
- Variants appeared quickly
- Created by 27-yr-old Filipino computer student Onel de Guzman
  - ❑ No local laws against spreading viruses
  - ❑ Creator given job as programmer! ☹

## 2000-06: Timofonica Worm

- E-mail enabled malware
- Automatically sent *pager* message to block of Telefonica cell phones
- Tried to delete all data on hard disk



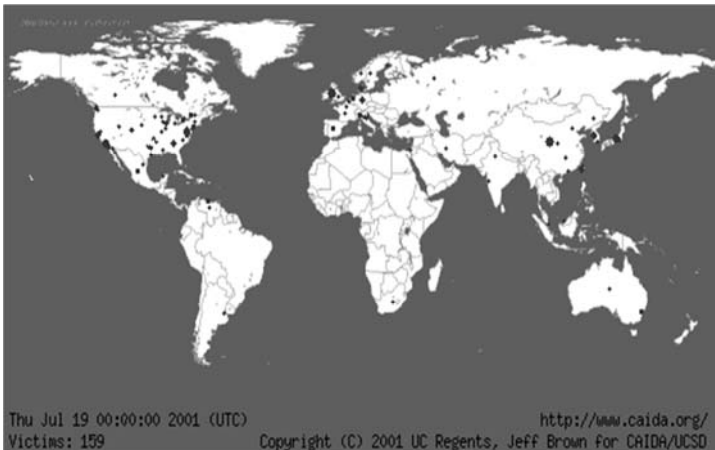
## 2001-03: SirCam Worm

- Propagated on Windows systems
- Used standard e-mail address books
- Infected document, converts to executable
  - ❑ Most naïve users turn off suffix display
  - ❑ So *myfile.doc.exe* looks like *myfile.doc*
- Created e-mail message with random subject and randomized text asking for comment
- Sent infected file to everyone on e-mail list
- Documents may contain confidential info
- See <http://www.cert.org/advisories/CA-2001-22.html>

## 2001-06: CodeRed Worm

- Infected vulnerable Web servers
  - ❑ Windows NT or Windows 2000 or CISCO equipment
  - ❑ running MS-IIS software that has not been *patched*
- Showed message on Web home page: HELLO! Welcome to <http://www.worm.com!> Hacked By Chinese!
- Sent copies of itself to computers in list of IP addresses
- On 20<sup>th</sup> through 28<sup>th</sup> of month, tried to swamp specific target with DoS (denial-of-service) attack
  - ❑ Original worm attacked numerical address of White House
  - ❑ Later versions received instructions from remote *master* computer program controlled by criminal hacker
- See <http://www.cert.org/advisories/CA-2001-19.html>

## Spread of the CodeRed Worm



## Trojans



- Named for the Trojan Horse in the Iliad
- Overt function useful or harmless
- Covert function unauthorized, usually harmful
- Functionality may be associated with all types of malware
  - ❑ Worms
  - ❑ Standalone programs
- Early Trojans included PC-Cyborg ("AIDS Information Disk") of 1989
  - ❑ Replaced autoexec.bat to count boots
  - ❑ On 90<sup>th</sup> boot, encrypted file/directory names
  - ❑ Author, Dr Joseph Popp arrested, extradited to US from UK, but never convicted due to mental incompetence



32

Copyright © 2011 M. E. Kabay. All rights reserved.

## 2000-01: Haiku Worm (Trojan)



F-Secure (formerly Data Fellows)

- E-mail enabled virus/worm
- Carrier: detailed e-mail message about Haiku generator
  - ❑ Actually works — Haiku in Windows box
- Worm code spreads through victim's e-mail address list
- Occasionally downloads and plays a .wav file from a Web site



33

Copyright © 2011 M. E. Kabay. All rights reserved.

## Spyware & Adware



- Software that collects user information without permission
  - ❑ Tracking & reporting Web usage
  - ❑ Monitoring use of licensed programs
  - ❑ Monitoring or blocking copying of music
  - ❑ Click-fraud (automatically clicks on ads for profit)
- Spyware serving unwanted ads = *adware*
- *Legal* issue is EULA (end-user license agreement)
  - ❑ If no clear statement of functions, spyware/adware may be violation of 18 US1030(a) (Computer Fraud & Abuse Act of 1986)
  - ❑ If EULA is clear and user agrees, matter of contract law
  - ❑ But many users never read EULA at all...
- Some spyware/adware difficult to uninstall (hides itself)

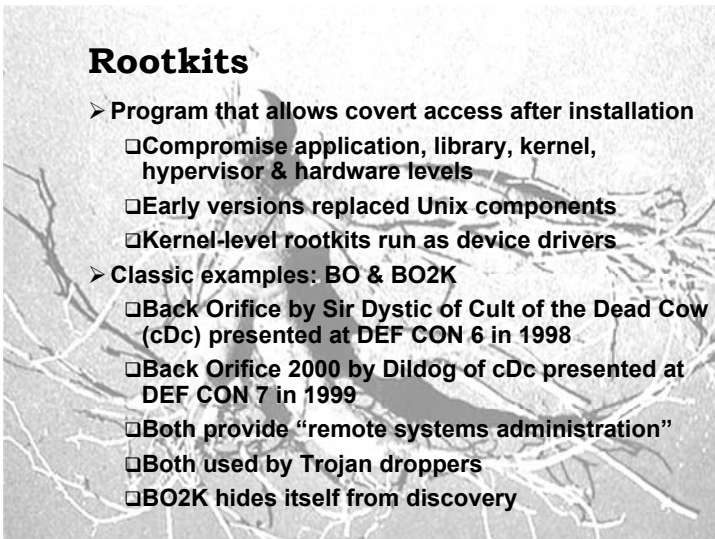


34

Copyright © 2011 M. E. Kabay. All rights reserved.

## Rootkits

- Program that allows covert access after installation
  - ❑ Compromise application, library, kernel, hypervisor & hardware levels
  - ❑ Early versions replaced Unix components
  - ❑ Kernel-level rootkits run as device drivers
- Classic examples: BO & BO2K
  - ❑ Back Orifice by Sir Dystic of Cult of the Dead Cow (cDc) presented at DEF CON 6 in 1998
  - ❑ Back Orifice 2000 by Dildog of cDc presented at DEF CON 7 in 1999
  - ❑ Both provide "remote systems administration"
  - ❑ Both used by Trojan droppers
  - ❑ BO2K hides itself from discovery



## Bots & Botnets (1)



- Bots
  - ❑ Automated processes on the Internet & WWW
  - ❑ Carry out specific tasks; e.g.,
    - ✓ Web spidering: collecting files from Web (e.g., GOOGLE engine bots)
    - ✓ Monitoring conversations on talk channels (e.g., for suppression of profanity or automated responses to questions)
- IRC Bots
  - ❑ Internet Relay Chat used for communications
  - ❑ IRC bots widespread for criminal activity
- Bot Herders control 100K bots for commercial (criminal) activity such as DDoS, spam



36

Copyright © 2011 M. E. Kabay. All rights reserved.

## Millions Of US Computers Completely Pwned By Botnets.

Oct 14, 2010

Categories: Business, Technology

by JJ SUTHERLAND

Microsoft just released its ninth Security Intelligence Report. The focus is on botnets. Microsoft cleaned 6.5 million computers from botnet infections in the three months April – June 2010. That's double the number a year ago. And remember, that's just those that have been cleaned. There are likely many millions more that haven't been.

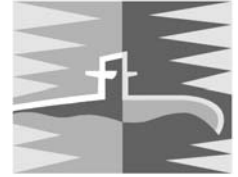
Country/Region	Computers with Bot Cleanings (1Q10)	Computers with Bot Cleanings (2Q10)	Bot Cleanings Per 1000 MSRT Executions (Bot CCM)
1 United States	2,163,216	2,148,169	5.2
2 Brazil	511,002	550,426	5.2
3 Spain	485,603	381,948	12.4
4 Korea	422,663	354,906	14.6
5 Mexico	364,554	331,434	11.4
6 France	344,743	271,478	4.0
7 United Kingdom	251,406	243,817	2.7

37

## Malicious Mobile Code



- Web servers host pages with active content
- Mobile code may be written in (e.g.)
  - ActiveX controls
  - Java applets
  - JavaScript
  - Adobe Flash
- Often involved in phishing attacks
- See *CSH5* Chapter 17, Mobile Code.



38

Copyright © 2011 M. E. Kabay. All rights reserved.

## Detection of Malicious Code



- Signature-Based
- Network-Based
- Behavioral
- Heuristic



39

Copyright © 2011 M. E. Kabay. All rights reserved.

## Signature-Based Malware Detection



- Oldest method of recognizing malware
  - Identify known strings of code/text
  - Defeated by polymorphism
- Hashes
  - Compute cryptographic hash of all executables on system; e.g.,
    - ✓ MD5
    - ✓ SHA-1
    - ✓ Digital signature using public key cryptosystem
  - Identify unauthorized changes (caused by malware) by checking table of hash values



40

Copyright © 2011 M. E. Kabay. All rights reserved.

## Network-Based Malware Detection



- Look for effects of running malware; e.g.,
  - Connection to unusual / characteristic server (like IRC)
  - Unusual protocols (not normal for system)
  - Peculiar packets (nor normal for protocol)
- Can establish baseline for behavior
  - Monitor KNOWN-CLEAN system
    - ✓ Critically important not to include malicious code in baseline
  - Compare observed behavior with baseline
  - Look for outliers & investigate deviations



41

Copyright © 2011 M. E. Kabay. All rights reserved.

## Behavioral Malware Detection



- Monitor behavior of code
  - Look for violations of security standards; e.g.,
    - Attempting to modify areas of memory outside local stack of process
    - Attempting to raise privilege level
- Sandboxes
  - Run code in restricted environment
  - E.g., Java sandbox
- Virtual machines a form of sandbox
  - Increasingly popular



42

Copyright © 2011 M. E. Kabay. All rights reserved.

## Heuristic Malware Detection

- *Heuristic* in this context means *able to change / learn*
- Apply statistical modeling & theoretical behavioral models
  - ❑ Computer score / metric to evaluate likelihood that program is legitimate
  - ❑ Can detect new variations of malware
    - ✓ Even if signature not yet registered by conventional scanners
- Modern antimalware products include option for heuristic scanning
  - ❑ Should enable it!



## Prevention of Malicious Code Attacks

- Defense in Depth vs Malware
- Operational Controls vs Malware
- Human Controls vs Malware
- Technical Controls vs Malware



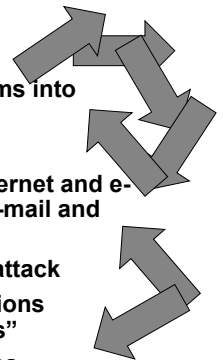
## Defense in Depth vs Malware

- No one AV program can protect against all malware
- Defense in depth uses multiple concurrent strategies
  - ❑ Operational controls
  - ❑ Human controls
  - ❑ Technical controls
- Different approach is to define *orthogonal* systems
  - ❑ Function in only one demonstrably correct way
  - ❑ But no one wants single-purpose, rigid systems



## Operational Controls vs Malware

- Written policies and procedures
  - ❑ Govern introduction of programs into production environment
  - ❑ Who can install programs?
  - ❑ Acceptable use policies for Internet and e-mail use (CSH5 Chapter 48, "E-mail and Internet Use Policies")
  - ❑ How to respond to suspected attack
  - ❑ See CSH5 Chapter 47, "Operations Security & Production Controls"
- Employment policies & procedures
  - ❑ CSH5 Chapter 45, "Employment Practices & Policies"



## Human Controls vs Malware

- Provide training on malware policies & procedures
- Topics
  - ❑ Current threats; e.g.,
    - ✓ Advance-fee fraud (Nigerian 419 fraud)
    - ✓ Social engineering (see CSH5 Chapter 29, "Social Engineering & Low-Tech Attacks")
    - ✓ Malicious attachments
  - ❑ Detecting the threats – not ignoring AV popups!
  - ❑ Proper response
    - ✓ Contact Help Desk at once



## Technical Controls vs Malware

- Implementing Antivirus Systems
- Host Configuration Controls & Security
- Network-Based Security Controls
- Network Monitoring



See CSH5 Chapter 41, "Antivirus Technology"

## Implementing Antivirus Systems



- Use both network-based & host-based systems
- Choose products from different vendors to run concurrently
- Run updates automatically on all systems daily *at least*
- E-mail may require separate appliance/system to control malware attachments, spam, fraud, phishing...



Copyright © 2011 M. E. Kabay. All rights reserved.

## Host Configuration Controls & Security



- Automatic updates / patches essential
- Eliminate non-critical software & services
  - ❑ Minimize threats that target growing complexity of environment
  - ❑ Current software development introduces average of 4.5 errors per 1000 lines of code
  - ❑ Inevitably, more code means more errors
  - ❑ Simplify environment to degree possible
- Browsers
  - ❑ Eliminate if possible
  - ❑ Otherwise, apply tight security
  - ❑ Use secure Web proxy



Copyright © 2011 M. E. Kabay. All rights reserved.

## Network-Based Security Controls



- Configure layered defense to interfere with malware propagation
  - ❑ Routers
  - ❑ Firewalls
  - ❑ Proxies
  - ❑ Switched virtual local area networks (VLANs)
- Filter aggressively
  - ❑ Bogus inbound network addresses (BOGONS)
    - ✓ Packet from an unassigned region of IP address space
  - ❑ Spoofed internal addresses
    - ✓ Claim to be from inside the target system
  - ❑ Packets from hostile countries (e.g., PRC) with whom you need no communications



Copyright © 2011 M. E. Kabay. All rights reserved.

## Network Monitoring



- Monitor & aggregate data from sensors
  - ❑ Device logs
  - ❑ Server logs
  - ❑ Host logs
  - ❑ Intrusion detection alerts
  - ❑ Network flow data
- Define historical database of normal behavior
- Look for anomalies – statistical outliers



Copyright © 2011 M. E. Kabay. All rights reserved.

## Awareness Tools



- ICSA Labs
  - ❑ <https://www.icsalabs.com/products>
- Virus Bulletin
  - ❑ <http://www.virusbntn.com/index>
- Avast!
  - ❑ <http://www.avast.com/virus-monitor>
- McAfee Virus Information
  - ❑ <http://home.mcafee.com/virusinfo>
- Microsoft Malware Protection Center
  - ❑ <http://www.microsoft.com/security/portal/>
- Sophos
  - ❑ <http://www.sophos.com/en-us/security-news-trends.aspx>
- Trend Micro
  - ❑ <http://us.trendmicro.com/us/trendwatch/>



Copyright © 2011 M. E. Kabay. All rights reserved.

## Current Situation (26 Sep 2011)

<http://us.trendmicro.com/us/trendwatch/>

**BREAKING NEWS - 23 September 2011**  
Dissecting the LURID APT - campaign, attacks, tactics and victims - read the report

### Virtualization and Cloud Computing

Security Threats to Evolving Data Centers

New Report Available >>

---

**Threat News**

<p><b>Security Advisories</b></p> <p>September 22, 2011 September 13, 2011</p> <p><b>Threats</b></p> <p>September 26, 2011 September 23, 2011</p> <p><b>Cloud Security</b></p> <p>September 23, 2011 September 8, 2011</p>	<p>Adobe Releases Out-of-Band Patch</p> <p>Five Important Bulletins for September Patch Tuesday</p> <p>Crooks Are Looking At Your Mobile Wallet</p> <p>LURID: Attribution Isn't Easy</p> <p>Join the Journey to the Cloud on Oct/5</p> <p>What Are the True Dangers of the Cloud?</p>
--	---

## Current Situation (26 Sep 2011)

TREND MICRO:  
<http://tinyurl.com/6evd7q6>

	Infected Files	Infected Computers	Risk
1 TROJ_WIMAD.AM	528,250	528,246	low
2 MAL_VJUNDO-4	489,148	488,822	low
3 CRYPT_NAIX-5	387,501	383,167	low
4 TROJ_VR_CEQ	262,850	262,841	low
5 PE_PATCHEP.A	235,085	233,705	low
6 MAL_OTORUNI	217,615	209,858	low
7 MAL_OTORUNI	155,557	154,156	low
8 TROJ_DLOADER.FXN	152,031	151,944	low
9 MAL_OTORUNI	137,314	99,700	low
10 MAL_NSANTI-1	137,137	132,107	low

TIME: 24 HOURS 1 WEEK 1 MONTH

55

Copyright © 2011 M. E. Kabay. All rights reserved.

## Current Situation (26 Sep 2011)

SANS Internet Storm Center:  
<http://isc.sans.edu/reports.html>

- Top 10 Ports (under attack)
- World Map with Country Reports
- Top 10 Source IP Addresses

Some recent NPR interviews with Alan Paller of SANS:

2010-07-19: <http://tinyurl.com/2633rxp>

2009-05-29: <http://tinyurl.com/5wdgrse>

56

Copyright © 2011 M. E. Kabay. All rights reserved.

# DISCUSSION

57

Copyright © 2011 M. E. Kabay. All rights reserved.