Malware

Fall 2024

R. Sekar

Current Threat Environment

- Steal confidential information
  - Credit-card/bank account #s, passwords, ...
  - Trade secrets and other proprietary information
  - Security-sensitive information
    - Useful for breaching physical world security
- Establish base for future operations
  - Conduit for future attacks
- Surveillance
  - Capture keystrokes, microphone or camera input
  - Reveal information about software installed
  - Snoop on web sites visited

Current Threats (Continued)

- Driven by commercialization of Malware
  - Thriving black-market for exploits
    - Zero-day exploits have arrived
  - "Bot"-centric model for cyber crime
    - Relay spam (e-mail scam, phishing)
    - Extortion (using DDoS or targeted attacks)
    - Focus on desktop (rather than server) vulnerabilities
  - Profit-driven adware and spyware
    - Customer-profiling, niche-marketing
    - IP protection (digital rights management)
    - aggressive installation, stealth (rootkits, spyware)

Current Threats (Continued)

- Targeted attacks on high-value targets
  - Political activists
  - International adversaries
  - People with access to valuable information
  - CEO/CFO with access to financial information on publicly traded companies
  - Researchers with access to proprietary formulas or other valuable IP

Types of Malware

- Viruses
- Worms
- DDoS and Botnet
- Rootkits
- Spyware
- ...

Computer Virus

- Properties
  - Replicates itself
  - Attaches to other non-malicious code
- Early versions spread via floppy disks, while recent viruses spread through the internet.
- Examples
  - Boot sector virus (difficult on OS with memory protection)
  - Other OS level virus
  - Virus that attaches to programs, scripts, libraries
  - Macro virus
  - Mail attachments
  - ...

Timeline of Notable Computer Viruses

- 1982: Elk Cloner (First virus in the wild, Apple II)
- 1986: Brain (First for IBM PC, a boot sector virus)
- 1995, Concept virus (first macro virus)
- 1998: CIH (very harmful, overwrites disk and BIOS)
- 1999: Melissa (Word/Outlook, floods email systems)
- 2000: ILOVEYOU (another email virus, estimated $10B losses)

Computer Worm

- Replicates over the network (usually by itself)
  - First worm appeared at Xerox PARC in 1978
- What a worm can do?
  - Replicates itself, and thus consumes network bandwidth
  - Deletes files on a host system
  - Sends documents via e-mail
  - Carries other executables as a payload
    - Installs a backdoor in an infected computer (zombie computer)
- Modern worms
  - Large scale infection
  - Fast spread rate
    - spread over the Internet within a second

Timeline of Notable Worms

- 1988: Morris worm (first well-known)
- 1999: Melissa (E-mail worm)
- 2000: ILOVEYOU (E-mail worm)
- 2001: Code Red (Exploited IIS bugs, slowed down the internet)
- 2003: Slammer (Exploited MS SQL Server bugs)
  - Very fast (75K victims in 10 minutes), very small (376 bytes!)
- 2003: Blaster, Welchia (Nachi), SoBig
- 2004: Sasser (Exploited LSASS bugs)
- 2007: Storm worm (Stealthy, established botnets)
  - Used obfuscation and rootkit-techniques to hide its behavior as well as its presence

Goals of Worms

- "bragging right" in early days
  - infect as many sites as possible
  - be as noticeable as possible
  - values fast spread, DoS effect
- Detection techniques could hence be targeted at these features
- More recently, worms used to establish botnets
  - Need to remain stealthy
    - Spread slowly so as to evade detection
    - Attacks launched on demand, but infection itself should not cause any noticeable surge in network traffic or other feature changes that can be easily spotted
    - So, we no longer hear about "high-profile" worms.

Rootkit

- Stealthy backdoor programs
- Intended to maintain "invisibility" of intruders
  - Intercepts data from terminals, network connections, and the keyboard
  - Conceals logins, running processes, files, logs, or other system data
- Origins of "rootkit"
  - Originally referred to such kind of programs in Unix systems (root - the administrator)

Rootkits

- Userlevel rootkits
  - Early ones on UNIX used to replace many programs used to examine system state
    - ls, ps, netstat,...
  - Drawback: if an administrator uses a custom C-program to examine system state, he can discover the presence of rootkit
- Kernel rootkits
  - System call interception based
    - All user level requests are intercepted and modified to hide the presence of rootkit
    - Problem: can be difficult to block all ways to learning about the presence of rootkit

More Advanced Rootkits

- May reside entirely within the kernel, with no user-level processes
- Hide themselves from system monitoring tools
  - e.g., put themselves on a scheduler queue, but not task queue
- In the most extreme case, avoid changing any data that is predictable or is read-only
  - Hide within kernel data structures that change all the time
- Rootkits that hide underneath the OS
  - Lift the OS into a VM!

Botnet

- A collection of compromised computers under a common control infrastructure
- Botnet's originator can control the group remotely
- Early botnets used means such as IRC
  - Stands out, and hence easier to spot
- Modern ones blend in
  - HTTP
  - Fast flux DNS
  - P2P

Botnet

- Purpose
  - DDoS
  - SMTP mail relays for SPAM
  - Theft of sensitive information
    - E.g. login IDs, credit card numbers, application serial numbers

Distributed Denial-of-Service (DDoS)

- DoS
  - An attack on a computer system or network that causes a loss of service to users
- Methods
  - Consumption of computational resources, such as bandwidth, disk space, or CPU time
  - Disruption of configuration information, such as routing information
  - Disruption of physical network components
- DDoS
  - Use of multiple hosts (often through Botnet) in a DoS

Spyware

- Properties
  - Intercept or take partial control of computer's operation
  - Without the informed consent of that computer's legitimate user.
  - Does not usually self-replicate.
- Purpose
  - Delivery of unsolicited pop-up advertisements
  - Theft of personal information
  - Monitoring of Web-browsing activity for marketing purposes
  - Routing of HTTP request to advertising sites

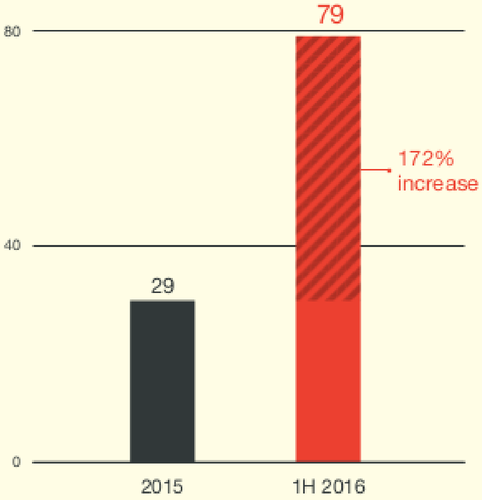
Spam

- Properties
  - Sending of unsolicited (commercial) emails
  - Sending nearly identical messages to thousands (or millions) of recipients
- Spamming in different media
- E-mail spam, Messaging spam, Newsgroup spam and Forum spam, Mobile phone spam, Internet telephony spam, Blog, wiki, guestbook, and referrer spam, etc

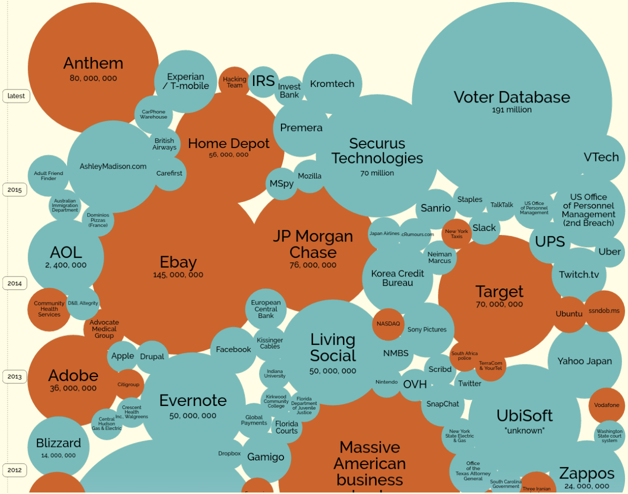
Phishing

- Uses social engineering techniques
  - Masquerading as a trustworthy person or business in an apparently official electronic communication
  - Attempts to fraudulently acquire sensitive information
    - Such as passwords and credit card details
- Spear-phishing
  - Phishing attack that is narrowly targeted at a single individual or a group of individuals

Ransomware . . .



Data Leaks . . .


latest
2015
2014
2013
2012
Anthem
80, 000, 000
Experian / T-mobile
Hacking Team
IRS
Invest Bank
Kromtech
Voter Database
191 million
CarPhone Warehouse
Home Depot
56, 000, 000
Premera
Securus Technologies
70 million
VTech
British Airways
AshleyMadison.com
Carefirst
Adult Friend Finder
Mozilla
MSpy
Australian Immigration Department
Dominios Pizzas (France)
Staples
TalkTalk
US Office of Personnel Management
US Office of Personnel Management (2nd Breach)
Sanrio
JP Morgan Chase
76, 000, 000
Japan Airlines
cRumours.com
New York Taxis
Slack
UPS
AOL
2, 400, 000
Ebay
145, 000, 000
Neiman Marcus
Uber
Korea Credit Bureau
Twitch.tv
Target
70, 000, 000
Community Health Services
D&B, Altegrity
European Central Bank
Ubuntu
ssndob.ms
Advocate Medical Group
NASDAQ
Sony Pictures
Living Social
50, 000, 000
Kissinger Cables
Facebook
Apple
Drupal
NMBS
South Africa police
TerraCom & YourTel
Yahoo Japan
Adobe
36, 000, 000
Citigroup
Indiana University
Scribd
Nintendo
OVH
Twitter
Evernote
50, 000, 000
Kirkwood Community College
Florida Department of Juvenile Justice
SnapChat
Vodafone
Crescent Health Inc. Walgreens
Central Hudson Gas & Electric
Global Payments
Florida Courts
UbiSoft
"unknown"
New York State Electric & Gas
Washington State court system
Blizzard
14, 000, 000
Dropbox
Gamigo
Massive American business
Office of the Texas Attorney General
South Carolina Government
Three Iranian
Zappos
24, 000, 000

Underlying Causes

- Untrusted software
  - Malware, including viruses, worms, bots, …
- Configuration errors
  - Default passwords, permissive firewall rules, …
- Human element
  - Insider threats, operator mistakes, social engineering
- Software vulnerabilities

Stealth and Obfuscation

- Malware wants to remain stealthy
  - So that it can be used in cyber crime (or to achieve other goals of attacker) without being detected
  - Protect "intellectual property"
- Intellectual property protection for legitimate code
  - Make it difficult to reverse-engineer code
  - Introduce watermarks
  - Prevent unauthorized copy of content
- Result
  - Obfuscation techniques

Evading Static Analysis Tools

- Low-level code obfuscation
  - Insert data in the middle of code
  - Violate typical ABI conventions, e.g., call/return, stack use, jumping to the middle of code, dynamic generation or modification code, etc.
  - Code encryption and transformation
- Higher level code obfuscation
  - Rename functions and variables
  - Control-flow obfuscation
- Data obfuscation

Control-flow Obfuscation

- Split or aggregate
  - Basic blocks
  - Loops
    - e.g., one loop becomes two loops or vice-versa
  - Procedures
    - Replace one procedure by two or merge two procedures
    - Inline a procedure, or outline (i.e., create new procedure)

Control-flow Obfuscation

- Reorder
- Insert dead-code (i.e., unreachable code)
  - Obfuscate using conditions
- Replace instruction sequences w/ alternate ones
- Insert conditional jumps using "opaque" predicates
- Insert indirect jumps
- Exploit aliasing and memory errors

Data Obfuscation

- Rename variables
- Split or aggregate variables
  - Split structures into individual variables or vice-versa
- Split individual variables
  - E.g., A = B - C - instead of A, use B and C
  - Clone a variable
- Pad arrays (and possibly structures) with junk elements
- “Encrypt” data values

Data Obfuscation

- Introduce extra levels of indirection
  - Instead of a simple variable, declare a pointer
- Introduce aliasing
- Introduce memory errors
- Introduce additional (or remove) function parameters

Evading Dynamic Analysis (Behavior obfuscation)

- Carefully match behavior with that of benign software, or employ code/behaviors that do not trigger suspicion
- Anti-analysis techniques
  - Detect execution within VM, emulator, or a sandbox and alter behavior
- Combine benign and malicious behaviors, complicating detection

Polymorphic viruses and encryption

- Historically, virus detection relied on “signatures” that captured byte sequences in code that were unique to the virus
- Polymorphism
  - Encrypt virus code so that it can change from one instance to another
  - Basically, change the encryption key from one generation to the next, causing massive changes to byte sequences
- Defense
  - Focus on invariant parts used to pack/unpack
  - Capture unpack/launch behavior (runtime detection)
  - Run virus scanner after unpack

Metamorphic Viruses

- Metamorphic viruses rewrite their entire code from one generation to next
- No "fixed" part in their code
  - Need not have any code encryption/decryption, so behavior based techniques can be defeated as well
- Metamorphic techniques
  - Use alternative instruction sequences to achieve the same effect
  - More general program obfuscation techniques

Key Issues in Malware Defense

- Plenty of motivation for attackers to remain stealthy
  - Many techniques are available to achieve this
    - Anti-virtualization, anti-analysis, obfuscation, ...
- Adaptive
  - Will employ evasion techniques specifically designed to defeat commonly deployed defenses

Key Issues in Malware Defense

- Need to assume a strong adversary model
  - Rely on self-protecting defense techniques
    - Ensure defense mechanisms are not compromised by malware
  - Complete mediation
  - Robustness against multi-step attacks ("stepping stones")