

# **MALICIOUS CODE**

**Information Security in Systems & Networks  
Public Development Program**

**Sanjay Goel**

**University at Albany, SUNY**

**Fall 2006**

# Malicious Code

## Learning Objectives

- Students should be able to:
  - Understand terminology & architecture of malicious code
  - Identify different types of malicious code
  - Recognize propagation mechanisms for malicious code
  - Determine controls for protection against malicious code
  - Realize financial impact of viruses & worms
  - Use the spread rates to compute the potential damage due to virus attacks

# Malicious Code

## Definition

- **Malicious code** –software that propagates through the network by exploiting weaknesses in software and protocols
- Basic types: Virus, Worm
- Variants of these basic types:
  - Trojan Horse
  - Time Bomb
  - Logic Bomb
  - Rabbit
  - Bacterium

# Malicious Code

## Virus: Definition

- **Definition:** Malicious self-replicating software that attaches itself to other software.
- **Typical Behavior:**
  - Replicates within computer system, potentially attaching itself to every other program
  - Behavior categories: Innocuous, Humorous, Data altering, Catastrophic

# Malicious Code

## Virus: Propagation

- Virus spreads by creating replica of itself and attaching itself to other executable programs to which it has write access.
  - True virus is NOT self-propagating and must be passed on to other users via e-mail, infected files/diskettes, programs or shared files
- Viruses normally consist of two parts:
  1. **Replicator:** responsible for copying the virus to other executable programs.
  2. **Payload:** Action of the virus, which may be benign such as printing a message or malicious such as destroying data or corrupting the hard disk.

# Malicious Code

## Virus:Process

- When a user executes an infected program (an executable file or boot sector), the replicator code typically executes first and then control returns to the original program, which then executes normally.
- Different types of viruses:
  - **Polymorphic viruses:** Viruses that modify themselves prior to attaching themselves to another program.
  - **Macro Viruses:** These viruses use an application macro language (e.g., VB or VBScript) to create programs that infect documents and template.

# Malicious Code

## Categories & Prevention

- **Vulnerabilities:** All computers
- **Common Categories:**
  - Boot sector
  - Terminate and Stay Resident (TSR)
  - Application software
  - Stealth (or Chameleon) / Mutation engine
  - Network
- **Prevention**
  - Limit connectivity
  - Limit downloads
  - Use only authorized media for loading data and software
  - Enforce mandatory access controls.
  - Viruses generally cannot run unless host application is running

# Malicious Code

## Detection

- Changes in file sizes or date/time stamps
- Computer is slow starting or slow running
- Unexpected or frequent system failures
- Change of system date/time
- Low computer memory or increased bad blocks on disks



# Malicious Code

## Detection Tools

- **Scanner** - a program that looks for known viruses by checking for recognizable patterns usually called signatures
  - Types: Conventional scanner, command-line scanner, on-demand scanner
- **Change Detectors** - programs that keep a database of the characteristics of all executable files on a system and check for changes which might signify an attack by an unknown virus.
  - Types: Check Summers & Integrity Checkers
- **Cryptographic Check Summers**
  - use an encryption algorithm to lessen the risk of being fooled by a virus which targets that particular check summer.

# Malicious Code

## Detection Tools Cont'd.

- **Monitor/Behavior Blocker** - a TSR that monitors programs while they are running for behavior which might denote a virus.
- **TSR scanner** - a TSR (memory-resident program) that checks for viruses while other programs are running. It may have some of the characteristics of a monitor and/or behavior blocker.
- **Heuristic Scanners** - scanners that inspect executable files for code using operations that might denote an unknown virus.

# Malicious Code

## Worms: Definition

- **Worms** –self-replicating programs like that can automatically spread.
  - Stand-alone applications
  - Do not need a carrier program
  - Replicate by spawning copies of themselves.
  - More complex and harder to write than the virus programs.
- **Vulnerabilities:** Multitasking computers, especially those employing open network standards

# Malicious Code: Worms and Variants

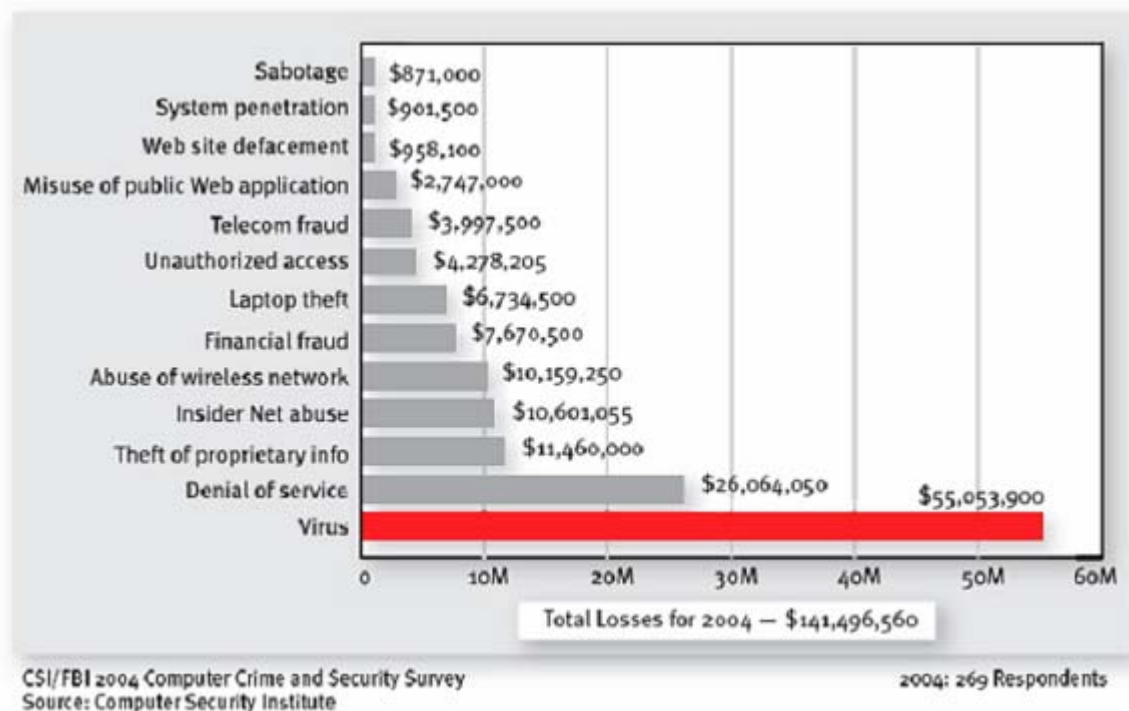
## Examples

- Internet Worm (1998)
  - First worm
  - Robert Morris Jr.
- ILOVEYOU worm (2000)
  - Automatically emailed itself to first 200 entries in Outlook address book
  - Spread to 10 million computers in two days before patch-release
  - Cost several billion dollars to repair the damage
- CodeRed, Nimbda, SirCam
  - Cost upwards of 500 million dollars in damages
- Anna Kournikova worm
  - Discovered August 2000
  - Became serious threat February 2001
  - Example of worm that took long time to spread

# Malicious Code: Worms and Variants

## Examples

- According to the CSI/FBI 2004 Computer Crime and Security Survey, the highest amount of loss was associated with viruses.



# Malicious Code: Worms and Variants

## Trojan Horse

- **Definition:** worm which pretends to be a useful program or virus purposely attached to a useful program prior to distribution
- **Typical Behaviors:** Same as virus or worm, but also can be used to send information back to or make information available to perpetrator
- **Vulnerabilities:**
  - Require user cooperation for executing their payload
  - Untrained users are vulnerable
- **Prevention:**
  - User training is best prevention (user cooperation allows Trojan Horses to bypass automated controls )
- **Detection:** Same as virus and worm

# Malicious Code

## Other Variants

- **Time Bomb**
  - A virus or worm designed to activate at a certain date/time
  - Behavior same as virus or worm, but widespread throughout organization upon trigger date
  - Time bombs are usually found before the trigger date
- **Logic Bomb**
  - A virus or worm designed to activate under certain conditions
- **Detection & Prevention**
  - Correlate user problem reports to find patterns indicating possible bomb
  - Run associated anti-viral software immediately as available

# Malicious Code

## Variants

- **Rabbit**
  - A worm designed to replicate to the point of exhausting computer resources
- **Bacterium**
  - Malicious code that forces the operating system to consume more and more CPU cycles, resulting eventually in noticeable delay in user transactions
  - Older versions of operating systems are more vulnerable than newer versions since hackers have had more time to write Bacterium
- **Detection**
  - Changes in OS file sizes, date/time stamps
  - Computer is slow in running
  - Unexpected or frequent system failures
- **Prevention**
  - Limit write privileges and opportunities to OS files
  - System administrators should work from non-admin accounts whenever possible.



# Malicious Code

## Counter Measure: Anti Virus

- Scan files and identify malicious code using virus (or worm) signatures
- Signatures are byte patterns extracted from the code of known virus
- Heuristic/pattern analysis (suspicious behavior) is also employed
- Approach works for virus, worms, spyware and adware

# Malicious Code

## Counter Measure: Firewall

- Classified based on different criteria
  - rule-based (filter based on ports, IP address, application, etc.)
  - hardware/software
  - network layer, application layer, application
  - personal, network based
  - stateless/stateful

# Malicious Code

## Counter Measure: Anti-Sypware

- Scan for changes to registry, browser default
- Search/home page, cookies, OS files, installed
- programs....
- Real-time/offline
- Based on signatures/unauthorized modification to the system

# Malicious Code

## Counter Measure: Intrusion Detection

- Software/hardware
- Based on network traffic or computer usage
- Includes sensor, console and central engine
- Signature/anomaly based
- Network/host based
- Passive/reactive
- Together with firewall to form IPS
- Examples
  - Snort (Network), Tripwire (Host)

# Malicious Code

## Summary

- Malicious code is software that spreads through networks that takes advantage of vulnerabilities in software code and protocols.
- Two major forms: worm and virus
- Viruses have two components a payload (exploit) and a replicator.
- Viruses need user intervention to spread.
- Worms are able to self-propagate throughout a network.
- Controls include anti-virus, proper firewall configuration, and various scanners.