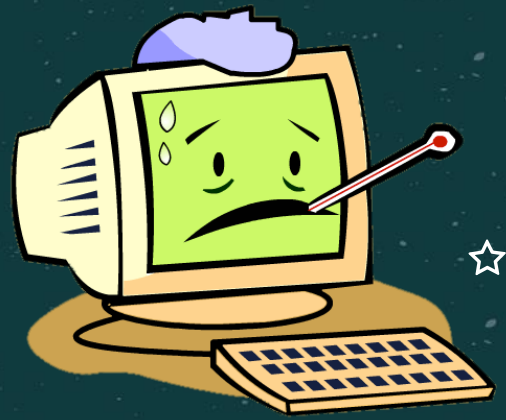
Malware - Viruses & Worms

Information Security - Lecture 22

Aadil Zia Khan

Virus

- Piece of software that infects programs
  - Modifies them to include a copy of the virus
  - Executes secretly when host program is run



Virus Structure

- Infection mechanism
  - This is how the virus propagates - typically has a search routine, which locates new files or new disks for infection
    - May include code which helps spread over the network
  - When the infected program is run, the virus code is also executed
- Trigger
  - This determines the event or condition for the malicious "payload" to be activated
- Payload
  - This is the actual code which carries out the malicious purpose of the virus
  - Payload activity might be noticeable (e.g., because it causes the system to slow down or "freeze"), or some times non-destructive but distributive, which is called virus hoax

Virus Life Cycle

- Dormant phase
  - The virus program has managed to access the target user's computer or software, but during this stage, the virus does not take any action and waits for the "trigger"

- Propagation phase
  - The virus starts propagating by placing a copy of itself into other programs or into certain system areas on the disk
  - The copy may be different from the original version - to avoid detection
  - Each infected program will now contain a clone of the virus, which will itself enter a propagation phase

Virus Life Cycle

- Triggering phase
  - A dormant virus moves into this phase when it is activated, and will now perform the function for which it was intended
  - The triggering phase can be caused by a variety of system events, e.g., number of copies crossing a threshold or a set period of days after an employee is fired, etc.

- Execution phase
  - This is the actual work of the virus, where the "payload" will be released
  - It can be destructive such as deleting files on disk, crashing the system, or corrupting files or relatively harmless such as popping up humorous or political messages on screen

Virus Classification By Target

- Boot sector infector
  - Infects a boot record and spreads when a system is booted from the disk containing the virus
  - Common because code in the boot sector is executed automatically
    - System BIOS often includes an option to prevent software from writing to the first sector of any attached hard drives to protect the master boot record
- File infector
  - Infects files that operating system or shell consider to be executable
- Macro virus
  - Infects files with macro code that is interpreted by an application

Virus Classification By Concealment Strategy

- Stealth techniques
  - Viruses are explicitly designed to hide itself from detection by antivirus software and humans
  - Some viruses (called cavity viruses) can infect files without increasing their sizes or damaging the files by overwriting unused areas of executable files
  - Some viruses make sure that the "last modified" date of a host file stays the same when the file is infected by the virus
  - Some viruses try to avoid detection by killing the tasks associated with antivirus software before it can detect them

Virus Classification By Concealment Strategy

- Encrypted virus
  - Virus creates a random encryption key, stored with the virus, and encrypts the remainder of the virus
  - When an infected program is invoked, the virus uses the stored random key to decrypt the virus
  - When the virus replicates, a different random key is selected

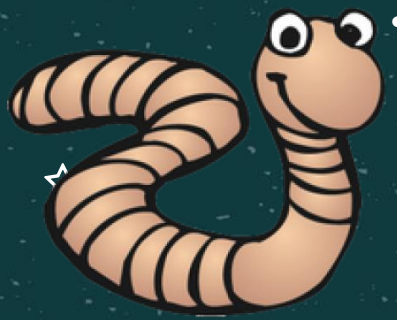
Virus Classification By Concealment Strategy

- Polymorphic virus
  - A virus that mutates with every infection, making detection by the “signature” of the virus impossible

- Metamorphic virus
  - As with a polymorphic virus ,a metamorphic virus mutates with every infection
  - The difference is that a metamorphic virus rewrites itself completely at each iteration, increasing the difficulty of detection
  - Metamorphic viruses may change their behavior as well as their appearance

Worm

- Worm - is a self-replicating program, similar to a computer virus
  - A virus attaches itself to, and becomes part of, another executable program; however, a worm is self-contained and does not need to be part of another program to propagate itself and attack



- Worm programs exploit software vulnerabilities in client or server programs to gain access to each new system
  - They can use network connections to spread or they can also spread through shared media, such as USB drives or CD and DVD data disks
  - E-mail worms can spread in macro or script code included in documents attached to e-mail or to instant messenger file transfers

Worm Target Discovery Strategy

- Random
  - Each compromised host probes random addresses in the IP address space
  - This technique produces a high volume of Internet traffic, causing disruption/detection even before the actual attack is launched

- Hit-List
  - The worm compiles a list of vulnerable machines over a long period to avoid detection before beginning infecting machines on the list

Worm Target Discovery Strategy

- Topological
  - Worm uses information contained in the infected victim machine to find more hosts to scan

- Local subnet
  - Infected hosts behind a firewall look for targets in its own local network using the subnet address structure

Payloads

- Stealthing
  - Backdoors
  - Rootkits

Payloads

- System corruption
  - Data destruction
  - Hardware destruction e.g., stuxnet
  - Ransom ware (encrypt data and ask money for the key)

Payloads

- Attack agent
  - DDoS
  - Spamming
  - Traffic sniffing
  - Keylogging
  - Spread malware
  - Online poll manipulation

Payloads

- Information theft / Spyware
  - Keyloggers
  - Credentials theft (username/passwords)
  - Identity theft (cnic/personal information)
  - Data theft (e.g., industrial espionage, government spying)
  - Phishing - Given sufficient details, the attacker can then “assume” the user’s identity for the purpose of obtaining credit, or sensitive access to other resources
    - Exploits social engineering to leverage user’s trust by masquerading as communications from a trusted source

Countermeasures - Antivirus

- Virus & antivirus tech have both evolved - early viruses simple code, easily removed - more complex now
- Antivirus over the years
  - A first-generation scanner requires a virus signature to identify a virus - structure and bit patterns (limited to the detection of known viruses)
  - A second-generation scanner uses heuristic rules to search for probable virus infection, e.g to look for fragments of code that are often associated with viruses
  - Third-generation scanner are memory-resident programs that identify a virus by its actions rather than structure in an infected program (e.g., interception of system calls)
  - Fourth-generation products include scanning, activity trap, and access control capability - limits the ability of viruses to penetrate a system and update files

Countermeasures - Sandbox

- Run potentially malicious code in an emulated sandbox or on a virtual machine
- These allow the code to execute in a controlled environment, where its behavior can be closely monitored without threatening the security of a real system.
    - Enables the detection of complex encrypted, polymorphic, or metamorphic malware
    - The code must transform itself into the required machine instructions before execution - resulting unpacked, transformed, or decrypted code can then be scanned for known malware signatures, or its behavior monitored as execution continues for possibly malicious activity