CS458: Introduction to Information Security

Notes 12: Malicious Software

Yousef M. Elmehdwi

Department of Computer Science

Illinois Institute of Technology

yelmehdwi@iit.edu

November 29th, 2018

Slides: Modified from Computer Security: Principles and Practice, 4th Edition. By: William Stallings and Lawrie Brown & Steven Gordon

Outline

Malicious Software (malware)

- Malware is "*a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim?s data, applications, or operating system or otherwise annoying or disrupting the victim*" - NIST

Classification of Malware

- Classified into two broad categories:
  - Propagation: how the malware spreads or propagates to reach the desired targets
    - Viruses
    - Worms
    - Social engineering
  - Payload: actions malware takes when reaches victim
    - System corruption
    - Zombies and bots
    - Information theft
    - Stealthing
- Countermeasures: anti-virus software

Types of Malicious Software (Malware)

- Propagation mechanisms include:
  - Infection of existing content by viruses that is subsequently spread to other systems
  - Exploit of software vulnerabilities by worms to allow the malware to replicate
  - Social engineering attacks that convince users to bypass security mechanisms to install Trojans or to respond to phishing attacks
- Payload actions performed by malware once it reaches a target system can include:
  - Corruption of system or data files
  - Theft of service/make the system a zombie agent of attack as part of a botnet
  - Theft of information from the system/keylogging
  - Stealthing/hiding its presence on the system

Outline

- Malicious Software
- Malware By Propagation Techniques
- Malware By Payloads
- Countermeasures
- Summary

Viruses

- A virus is piece of software that “infects” programs and copies itself to other programs
  - Modifies them to include a copy of the virus
  - Replicates and goes on to infect other content
  - Easily spread through network environments
- When attached to an executable program a virus can do anything that the program is permitted to do
  - Executes secretly when the host program is run
- Most viruses are specific to operating systems and/or hardware platforms
  - Takes advantage of their details and weaknesses

Virus Components

- Infection mechanism
  - Means by which a virus spreads or propagates
  - Also referred to as the *infection vector*
- Trigger
  - Event or condition that determines when the payload is activated or delivered Sometimes known as a *logic bomb*
- Payload
  - What the virus does (besides spreading)
  - May involve damage or benign but noticeable activity

Virus Phases

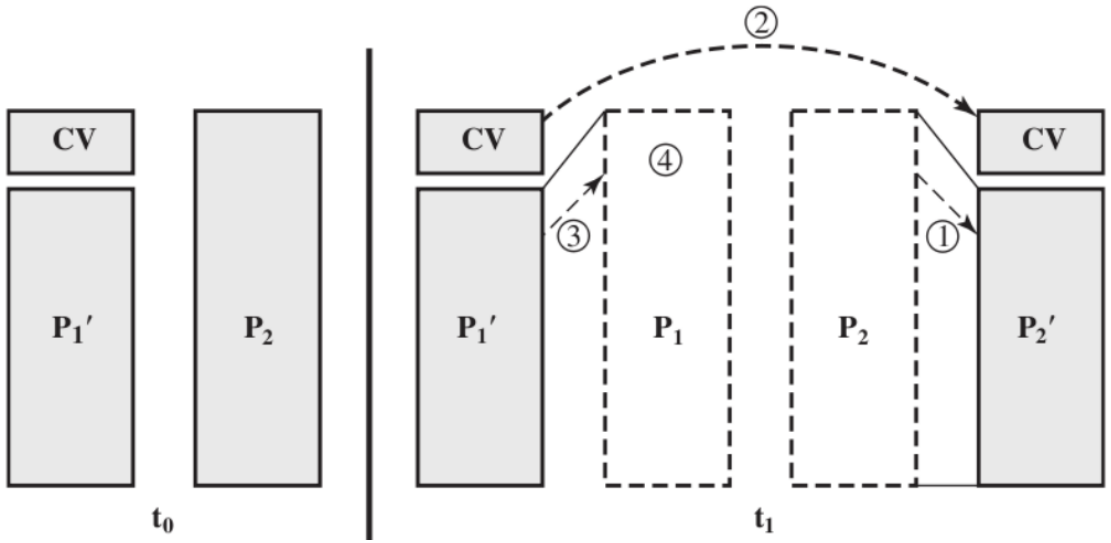
- During its lifetime, a typical virus goes through the following four phases:
  1. Dormant
     - virus is idle; will be activated by some event (like logic bomb)
     - Not all viruses have this stage
  2. Propagation
     - Virus places a copy of itself into other programs or into certain system areas on the disk
     - May not be identical to the propagating version
     - Each infected program will now contain a clone of the virus which will itself enter a propagation phase
  3. Triggering
     - virus is activated to perform the function for which it was intended
     - can be caused by a variety of system events, e.g., count of the number of times virus has made copies of itself.
  4. Execution
     - function is performed, either harmless (display a message) or malicious (delete or modify files)

A Simple Virus

```
program V :=
{goto main;
    1234567;

    subroutine infect-executable :=
        {loop:
        file := get-random-executable-file;
        if (first-line-of-file = 1234567)
             then goto loop
        else
             prepend V to file; }

    subroutine do-damage :=
        {whatever damage is to be done}

    subroutine trigger-pulled :=
        {return true if some condition holds}

main: main-program :=
    {infect-executable;
    if trigger-pulled
          then do-damage;
    goto next;}
next:
}
```

Compression Virus

- The simple virus can be detected because file length is different from original program
- This detection can be avoided using compression
- Assume program P_1' is infected with virus CV
  1. For each uninfected file P_2, the virus compresses P_2 to produce P_2'
  2. Virus CV is pre-pended to P_2' (so resulting size is same as P_2)
  3. P_1' is uncompressed and (4) executed



A Compression Virus

```
program CV :=
{  goto main;
   01234567;

   subroutine infect-executable :=
       { loop:
          file := get-random-executable-file;
          if (first-line-of-file = 01234567)
              then goto loop
          (1) compress file;
          (2) prepend CV to file;
         }

  main: main-program :=
    {  if ask-permission
            then infect-executable;
       (3) uncompress rest-of-file;
       (4) run uncompressed file;
    }
}
```

Virus Classifications

- Classification by target
  - the type of target the virus tries to infect
- Classification by concealment strategy
  - the method the virus uses to conceal itself from detection by users and anti-virus software

Classification by target

A virus classification by target includes the following categories:

- Boot Sector Infector infects a master boot record or boot record and spreads when a system is booted from the disk containing the virus
- File Infector infects files that the operating system or shell considers to be executable
- Macro Virus infects files with macro or scripting code that is interpreted by an application
- Multipartite Virus infects files in multiple ways
  - capable of infecting multiple types of files

Classification by concealment strategy

- Encrypted Virus a portion of the virus creates a random encryption key and encrypts the remainder of the virus
- Stealth Virus a form of virus explicitly designed to hide itself from detection by anti-virus software. Thus, the entire virus, not just a payload is hidden.
- Polymorphic Virus a form of virus that creates copies during replication that are functionally equivalent but have distinctly different bit patterns, in order to defeat programs that scan for viruses (a virus that mutates with every infection)
- Metamorphic Virus a virus that mutates and rewrites itself completely at each iteration and may change behavior as well as appearance

Macro and Scripting Viruses

- "a virus that attaches itself to documents and uses the macro programming capabilities of the document's application to execute and propagate"
- Macro viruses infect scripting code used to support active content in a variety of user document types
- Are threatening for a number of reasons:
  - Is platform independent
  - Infect documents, not executable portions of code
  - Are easily spread
  - Because they infect user documents rather than system programs, traditional file system access controls are of limited use in preventing their spread, since users are expected to modify them
  - Are much easier to write or to modify than traditional executable viruses

Worms

Malware propagation that concerns the exploit of software vulnerabilities which are commonly exploited by computer worms.

- Program that actively seeks out more machines to infect and each infected machine serves as an automated launching pad for attacks on other machines
- Exploits software vulnerabilities in client or server programs
- Can use network connections to spread from system to system
- Spreads through shared media (USB drives, CD, DVD data disks)
- E-mail worms spread in macro or script code included in attachments and instant messenger file transfers
- Upon activation the worm may replicate and propagate again
- Usually carries some form of payload

Worm Replication

To replicate itself, a worm uses some means to access remote systems, include the following:

- E-mail or instant messaging worm e-mails a copy of itself to other systems; sends itself as an attachment via an instant message service
- File sharing creates a copy of itself or infects a file as a virus on removable media
- Remote execution capability worm executes a copy of itself on another system
- Remote file access capability worm uses a remote file access or transfer service to copy itself from one system to the other
- Remote login capability worm logs onto a remote system as a user and then uses commands to copy itself from one system to the other

Social Engineering

- Tricking users to assist in the compromise of own system
- This can occur when a user views and responds to some SPAM e-mail, or permits the installation and execution of some Trojan horse program or scripting code.
- Spam Email
  - Unsolicited bulk email
  - Common carrier of malware as attachments or via links
  - Used for phishing attacks
- Trojan Horses
  - Useful software that also performs harmful functions
  - Used to accomplish functions that the attacker could not accomplish directly
- Mobile phone Trojans
  - First appeared in 2004
  - Target is the smartphone

Outline

- Malicious Software
- Malware By Propagation Techniques
- Malware By Payloads
- Countermeasures
- Summary

System Corruption

Once malware is active on the target system, the next concern is what actions it will take on this system. That is, what payload does it carry.

- Action taken by malware on system: corrupt the system
- Data Destruction delete, overwrite data; encrypt data and then demand payment to decrypt (ransomware)
- Real-World Damage
  - Causes damage to physical equipment (*Chernobyl virus* rewrites BIOS code)
  - *Stuxnet worm*[1] Targets specific industrial control system software
  - There are concerns about using sophisticated targeted malware for industrial sabotage
- Logic Bomb
  - key component of data corrupting malware
  - Code embedded in the malware that is set to "explode" when certain conditions are met, e.g. presence/absence of files, data/time, particular software or user

[1] W32.Stuxnet Dossier

Ransomware: WannaCry

- Infected a large number of systems in many countries in May 2017
- When installed on infected systems, it encrypted a large number of files and then demanded a ransom payment in Bitcoins to recover them
- Recovery of this information was generally only possible if the organization had good backups and an appropriate incident response and disaster recovery plan
- Targets widened beyond personal computer systems to include mobile devices and Linux servers
- Tactics such as threatening to publish sensitive personal information, or to permanently destroy the encryption key after a short period of time, are sometimes used to increase the pressure on the victim to pay up

Attack Agent Bots

- Takes over another Internet attached computer and uses that computer to launch or manage attacks
  - Such a system is known as a bot (robot), zombie or drone
- Infected machines are bots. Victim is unaware of infection
- botnet: collection of bots capable of acting in a coordinated ("network" of infected machines)
- This type of payload attacks the integrity and availability of the infected system
- Uses of Bots
  - distributed denial-of-service (DDoS) attacks
  - spamming
  - sniffing traffic
  - keylogging
  - spreading new malware
  - installing advertisement add-ons and browser plugins
  - manipulating online polls/games: Since every bot has a distinct IP address, every vote will have the same credibility as a vote cast by a real person

Information Theft

payloads where the malware gathers data stored on the infected system for use by the attacker

- Keyloggers
  - Captures keystrokes to allow attacker to monitor sensitive information
  - Typically uses some form of filtering mechanism that only returns information close to keywords, e.g. “login”, “password”
- Spyware
  - Subverts the compromised machine to allow monitoring of a wide range of activity on the system
  - Monitoring history and content of browsing activity
  - Redirecting certain Web page requests to fake sites
  - Dynamically modifying data exchanged between the browser and certain Web sites of interest

Information Theft

Another approach used to capture a user's login and password credentials

- Phishing
  - Exploits social engineering to leverage the user's trust by masquerading as communication from a trusted source
  - Include a URL in a spam e-mail that links to a fake Web site that mimics the login page of a banking, gaming, or similar site
  - Suggests that urgent action is required by the user to authenticate their account
  - Attacker exploits the account using the captured credentials
  - Spear-phishing
    - recipients are carefully researched by the attacker (targeted phishing)
    - e-mail is crafted to specifically suit its recipient, often quoting a range of information to convince them of its authenticity

Stealthing

Techniques used by malware to hide its presence on the infected system, and to provide covert access to that system.

- Backdoor. Also known as a trapdoor
  - Secret entry point into a program allowing the attacker to gain access and bypass the security access procedures
- Rootkit
  - Set of hidden programs installed on a system to maintain covert access to that system
  - Hides by subverting the mechanisms that monitor and report on the processes, files, and registries on a computer
  - Gives administrator (or root) privileges to attacker. Can add or change programs and files, monitor processes, send and receive network traffic, and get backdoor access on demand

Outline

- Malicious Software
- Malware By Propagation Techniques
- Malware By Payloads
- Countermeasures
- Summary

Malware Countermeasure Approaches

- Ideal solution to the threat of malware is prevention
  - Elements of prevention: policy, awareness, vulnerability mitigation, threat mitigation
  - Ensure systems are up-to-date, patches applied
  - Apply access controls
  - User awareness and training
- If prevention fails, technical mechanisms can be used to support the following threat mitigation options:
  - Detection, identification and removal

Malware Countermeasure Approaches

- Requirements for effective malware countermeasures
  - **Generality**: should be able to handle a wide variety of attacks.
  - **Timeliness**: should respond quickly so as to limit the number of infected programs or systems and the consequent activity
  - **Resiliency**: should be resistant to evasion techniques employed by attackers to hide the presence of their malware.
  - **Minimal denial-of-service costs**: should result in minimal reduction in capacity or service due to the actions of the countermeasure software, and should not significantly disrupt normal operation.
  - **Transparency**: countermeasure software and devices should not require modification to existing (legacy) OSs, application software, and hardware.
  - **Global and local coverage**: should be able to deal with attack sources both from outside and inside the enterprise network.
- Multiple approaches to meet all these requirements. Detection of the presence of malware can occur in a number of locations
  - Host-based scanners, perimeter scanning, distributed intelligence gathering

Development of Anti-virus Software

- 1st generation: simple scanners
  - Requires a malware signature to identify the malware
  - Limited to the detection of known malware
- 2nd generation: heuristic scanners
  - Uses heuristic rules to search for probable malware instances
  - Another approach is integrity checking (checksum can be appended to each program)
- 3rd generation: activity traps
  - Memory-resident programs that identify malware by its actions rather than its structure in an infected program
- 4th generation: full-featured protection
  - Packages consisting of a variety of anti-virus techniques used in conjunction
  - Include scanning and activity trap components and access control capability

Generic Decryption

- A polymorphic virus must decrypt itself to activate
- Generic decryption runs executable code in virtual machine, monitors instructions
  - CPU emulator: virtual machine software
  - Virus signature scanner: scans for signatures
  - Emulation control module: controls execution of target code
- If decryption performed, malware is exposed and detected
- Enables anti-virus program to easily detect complex polymorphic viruses and other malware while maintaining fast scanning speeds
- How long to run each interpretation?
  - Too long: system performance degraded
  - Too short: do not see malware

Host-Based Behavior Blocking Software

- Unlike heuristics or fingerprint-based scanners, dynamic malware analysis or behavior-blocking software integrates with OS, monitors program behavior in real-time
- Block potentially malicious actions before they affect system
  - Attempts to open, view, delete, modify files
  - Attempts to format disks
  - Modifications to logic of executable files
  - Modification of critical system settings
  - Scripting of email or IM clients to send executable files
  - Initiation of network connections
- Doesn't depend on signatures or fingerprinting
- Limitations: Allows malicious code to run, some actions may be undetected

Outline

- Malicious Software
- Malware By Propagation Techniques
- Malware By Payloads
- Countermeasures
- Summary

Key Points

- Many types of malware
- Virus infects content, propagate attached to files
- Worms exploit software vulnerabilities to distribute itself
- Social engineering used to trick users into performing harmful actions
- Malware payloads may destruct data and damage physical objects
- Anti-virus software continues to develop, using multiple approaches

Security Issues

- Cat-and-mouse: many countermeasures rely on knowledge of existing malware, malware producers try to defeat countermeasures
- Performance degradation and denial-of-service: countermeasures often affect normal system behavior
- What can you trust?