Malicious Software

ITS335: IT Security

Sirindhorn International Institute of Technology
Thammasat University

Prepared by Steven Gordon on 20 December 2015
its335y15s2l05, Steve/Courses/2015/s2/its335/lectures/malicious.tex, r4287

Contents

Malicious Software

Malware By Propagation Techniques

Malware By Payloads

Countermeasures

Summary

Malicious Software

- 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
- A classification of malware:

  Propagation how the malware spreads
    - Viruses
    - Worms
    - Social engineering

  Payload actions malware takes when reaches victim
    - System corruption
    - Zombies and bots
    - Information theft
    - Stealthing
- Countermeasures: anti-virus software

Contents

Malicious Software

Malware By Propagation Techniques

Malware By Payloads

Countermeasures

Summary

Nature of Viruses

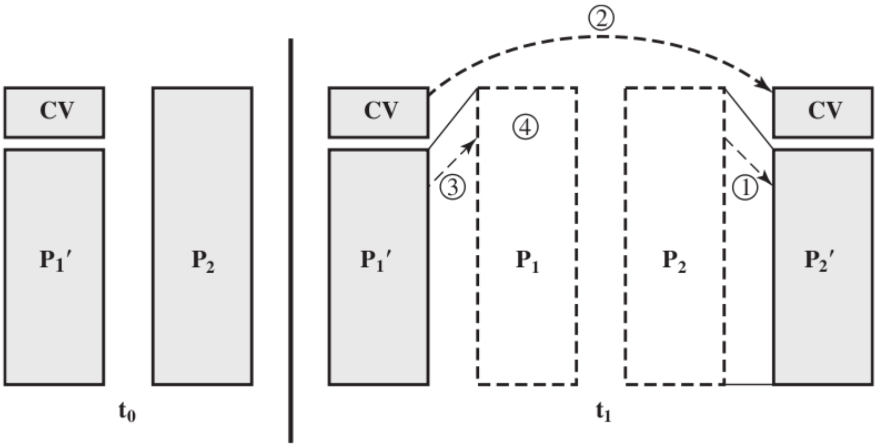
- A virus is piece of software that "infects" programs and copies itself to other programs
- The phases of a virus are:
  1. Dormant: virus is idle; will be activated by some event (like logic bomb)
  2. Propagation: virus copies itself into other programs or areas of operating system
  3. Triggering: virus is activated to perform some function; similar triggers to logic bombs, but also number of times virus copied
  4. Execution: function is performed, either harmless (display a message) or malicious (delete or modify files)
- Most viruses are specific to operating systems and/or hardware platforms

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 P1 is infected with virus CV
  1. For each uninfected file P2, the virus compresses P2 to produce P2
  2. Virus CV is pre-pended to P2 (so resulting size is same as P2)
  3. P1 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;}
}
```

Types of Viruses: By Target

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

Types of Viruses: 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

Polymorphic Virus a virus that mutates with every infection

Metamorphic Virus a virus that mutates and rewrites itself completely at each iteration and may change behaviour as well as appearance

Example Viruses

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

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

Example Worms

Social Engineering

Tricking users to assist in the compromise of own system

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

Contents

Malicious Software

Malware By Propagation Techniques

Malware By Payloads

Countermeasures

Summary

System Corruption

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 corrupt BIOS code so computer cannot boot; control industrial systems to operate such that they fail, e.g. Stuxnet worm

Logic Bomb activate when certain conditions are met, e.g. presence/absence of files, data/time, particular software or user

Zombies and Bots

- Take over another Internet attached computer and uses that computer to launch or manage attacks
- botnet: collection of bots capable of acting in a coordinated manner
- Uses:
  - distributed denial-of-service (DDoS) attacks
  - spamming
  - sniffing traffic
  - keylogging
  - spreading new malware
  - installing advertisement add-ons and browser plugins
  - attacking IRC chat networks
  - manipulating online polls/games

Information Theft

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

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
  - e-mail is crafted to specifically suit its recipient, often quoting a range of information to convince them of its authenticity

Other Malware

- Backdoor
- Trapdoor
- Rootkit
- Mobile code
- Drive-by-downloads
- Flooders
- . . .

Contents

Malicious Software

Malware By Propagation Techniques

Malware By Payloads

Countermeasures

Summary

Malware Countermeasure Approaches

- Prevention is ideal solution, but almost impossible
  - Elements of prevention: policy, awareness, vulnerability mitigation, threat mitigation
  - Ensure systems are up-to-date, patches applied
  - Apply access controls
  - User awareness and training
- Detection, identification and removal
- Requirements of countermeasures:
  - Generality, timeliness, resiliency, minimal denial-of-service costs, transparency, global and local coverage
- Multiple approaches to meet requirements:
  - 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

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 Behaviour Blocking Software

- Integrates with OS, monitors program behaviour 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
- Allows malicious code to run, some actions may be undetected

Contents

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 behaviour
- What can you trust?

Areas To Explore

- Trusted computing
- Digital espionage and cyber-warfare
- Malware on mobile devices
- Perimeter scanning and Digital Immune System