COMPUTER SECURITY

PRINCIPLES AND PRACTICE

SECOND EDITION

William Stallings | Lawrie Brown

Chapter 6

Malicious Software

Malware

[NIST05] defines malware as:

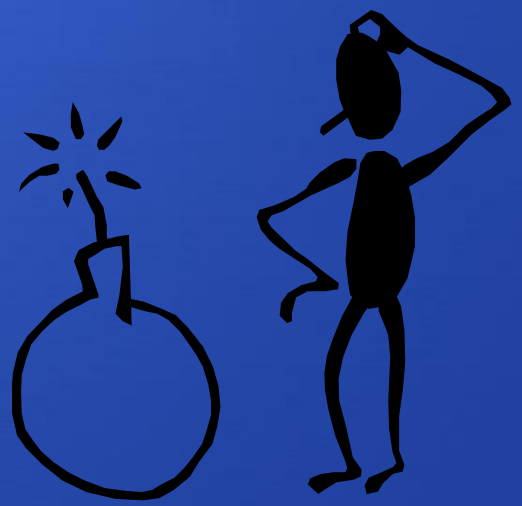
"a program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim."

Name	Description
Adware	Advertising that is integrated into software. It can result in pop-up ads or redirection of a browser to a commercial site.
Attack Kit	Set of tools for generating new malware automatically using a variety of supplied propagation and payload mechanisms
Auto-rooter	Malicious hacker tools used to break into new machines remotely.
Backdoor (trapdoor)	Any mechanisms that bypasses a normal security check; it may allow unauthorized access to functionality in a program, or onto a compromised system.
Downloaders	Code that installs other items on a machine that is under attack. It is normally included in the malware code first inserted on to a compromised system to then import a larger malware package.
Drive-by-Download	An attack using code in a compromised web site that exploits a browser vulnerability to attack a client system when the site is viewed.
Exploits	Code specific to a single vulnerability or set of vulnerabilities.
Flooders (DoS client)	Used to generate a large volume of data to attack networked computer systems, by carrying out some form of denial-of-service (DoS) attack.
Keyloggers	Captures keystrokes on a compromised system.
Logic bomb	Code inserted into malware by an intruder. A logic bomb lies dormant until a predefined condition is met; the code then triggers an unauthorized act.
Macro Virus	A type of virus that uses macro or scripting code, typically embedded in a document, and triggered when the document is viewed or edited, to run and replicate itself into other such documents.
Mobile code	Software (e.g., script, macro, or other portable instruction) that can be shipped unchanged to a heterogeneous collection of platforms and execute with identical semantics.
Rootkit	Set of hacker tools used after attacker has broken into a computer system and gained root-level access.
Spammer programs	Used to send large volumes of unwanted e-mail.
Spyware	Software that collects information from a computer and transmits it to another system by monitoring keystrokes, screen data and/or network traffic; or by scanning files on the system for sensitive information.
Trojan horse	A computer program that appears to have a useful function, but also has a hidden and potentially malicious function that evades security mechanisms, sometimes by exploiting legitimate authorizations of a system entity that invokes the Trojan horse program.

Table 6.1

Virus	Malware that, when executed, tries to replicate itself into other executable machine or script code; when it succeeds the code is said to be infected. When the infected code is executed, the virus also executes.
Worm	A computer program that can run independently and can propagate a complete working version of itself onto other hosts on a network, usually by exploiting software vulnerabilities in the target system.
Zombie, bot	Program activated on an infected machine that is activated to launch attacks on other machines.

Malware Terminology



Classification of Malware

classified into two broad categories:

- based first on how it spreads or propagates to reach the desired targets
- then on the actions or payloads it performs once a target is reached

also classified by:

- those that need a host program (parasitic code such as viruses)
- those that are independent, self-contained programs (worms, trojans, and bots)
- malware that does not replicate (trojans and spam e-mail)
- malware that does replicate (viruses and worms)

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 or drive-by-downloads 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

Viruses

- piece of software that infects 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
- specific to operating system and hardware
  - 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

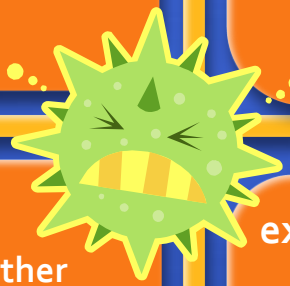
Virus Phases

dormant phase

- virus is idle
- will eventually be activated by some event
- not all viruses have this stage

triggering phase

- virus is activated to perform the function for which it was intended
- can be caused by a variety of system events



propagation phase

- 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

execution phase

- function is performed
- may be harmless or damaging

Virus Structure

```
    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:

}
```

Figure 6.1 A Simple Virus

Compression Virus Logic

```
    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;}
       }
```

Figure 6.2 Logic for a Compression Virus

Operation for Figure 6.2

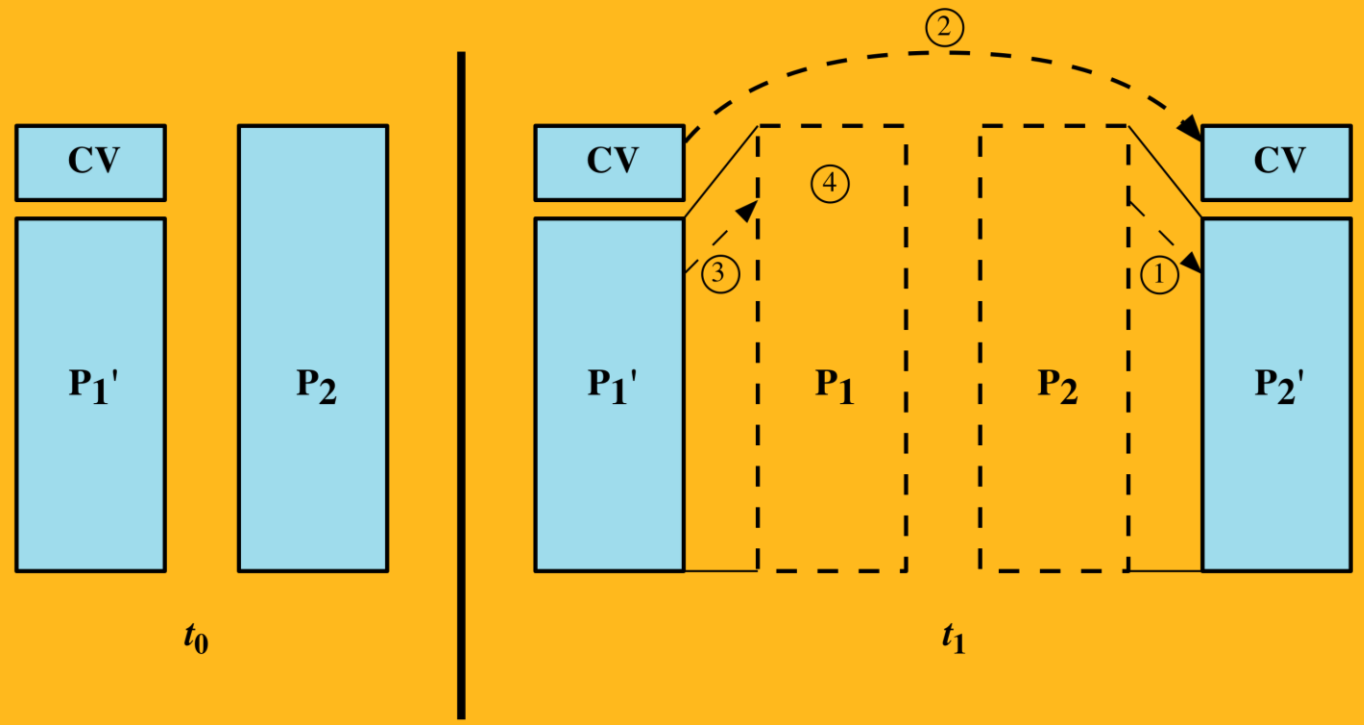


Figure 6.3 A Compression Virus

Virus Classifications

classification by target

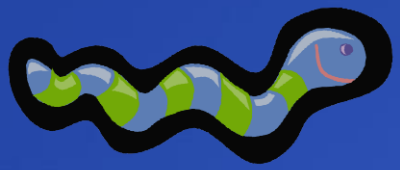
- boot sector infector
  - infects a master boot record or boot record and spreads when a system is booted from the disk containing the virus
- file infector
  - infects files that the operating system or shell considers to be executable
- macro virus
  - infects files with macro or scripting code that is interpreted by an application
- multipartite virus
  - infects files in multiple ways

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
- 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 behavior as well as appearance

Macro/Scripting Code Viruses

- very common in mid-1990s
  - platform independent
  - infect documents (not executable portions of code)
  - easily spread
- exploit macro capability of MS Office applications
  - more recent releases of products include protection
- various anti-virus programs have been developed so these are no longer the predominant virus threat



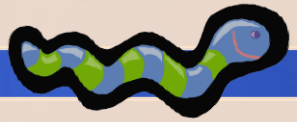
Worms

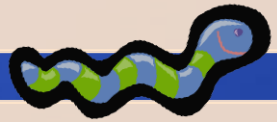
- program that actively seeks out more machines to infect and each infected machine serves as an automated launching pad for attacks on other machines
- exploits software vulnerabilities in client or server programs
- can use network connections to spread from system to system
- spreads through shared 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
- first known implementation was done in Xerox Palo Alto Labs in the early 1980s

Worm Replication



electronic mail or instant messenger facility	• 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 or transfer 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





Worm Propagation Model

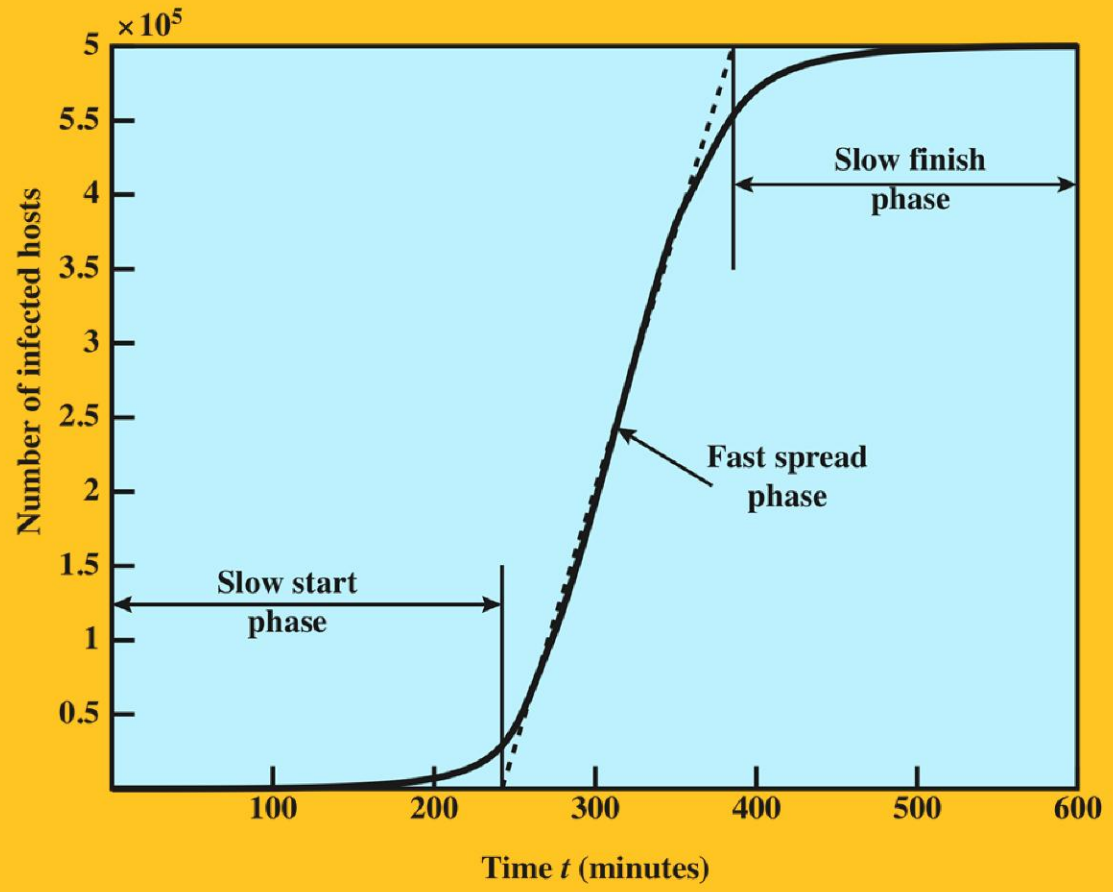


Figure 6.4 Worm Propagation Model

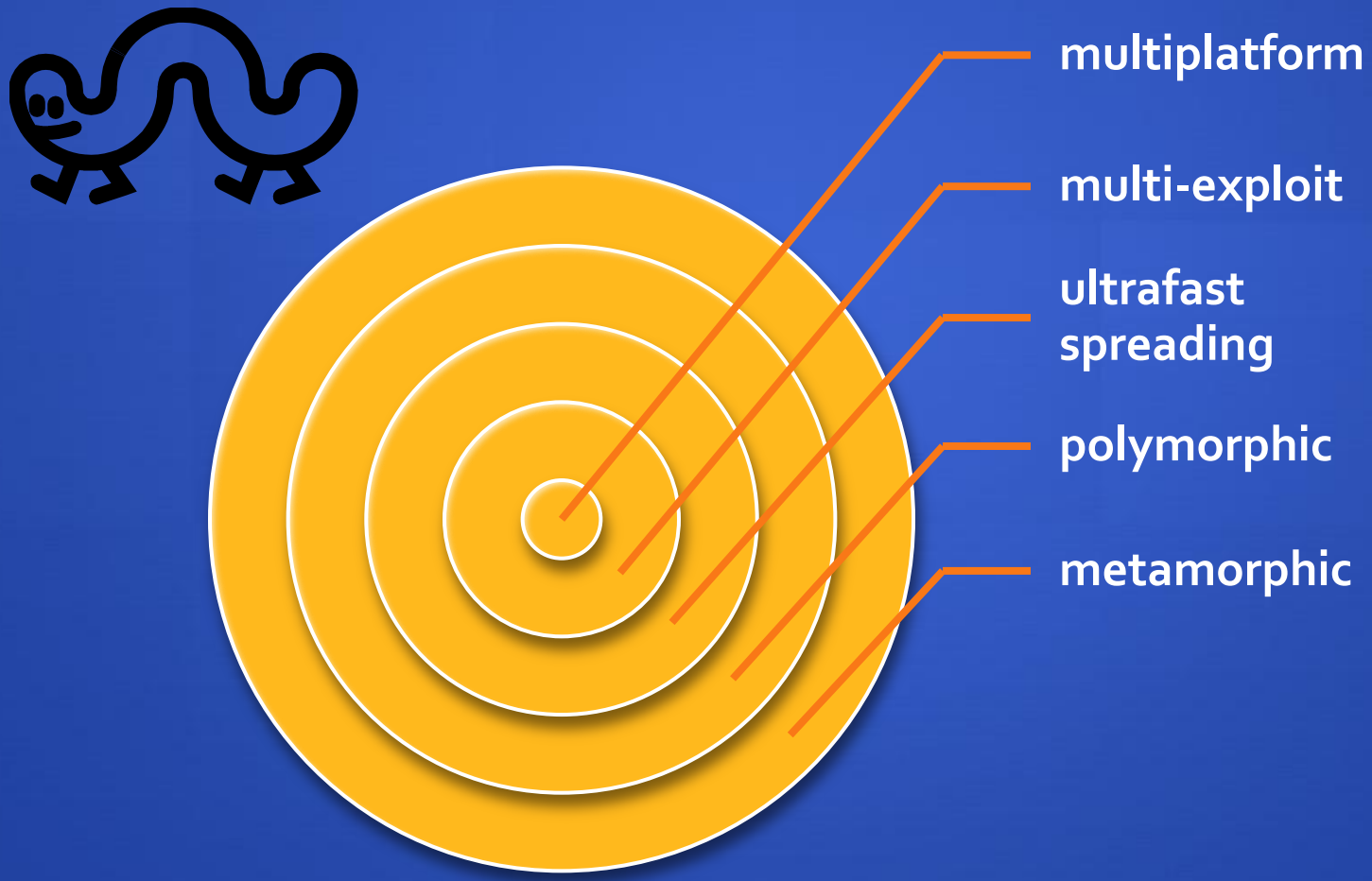
Morris Worm

- earliest significant worm infection
- released by Robert Morris in 1988
- designed to spread on UNIX systems
  - attempted to crack local password file to use login/password to logon to other systems
  - exploited a bug in the finger protocol which reports the whereabouts of a remote user
  - exploited a trapdoor in the debug option of the remote process that receives and sends mail
- successful attacks achieved communication with the operating system command interpreter
  - sent interpreter a bootstrap program to copy worm over

Recent Worm Attacks

Melissa	1998	e-mail worm first to include virus, worm and Trojan in one package
Code Red	July 2001	exploited Microsoft IIS bug probes random IP addresses consumes significant Internet capacity when active
Code Red II	August 2001	also targeted Microsoft IIS installs a backdoor for access
Nimda	September 2001	had worm, virus and mobile code characteristics spread using e-mail, Windows shares, Web servers, Web clients, backdoors
SQL Slammer	Early 2003	exploited a buffer overflow vulnerability in SQL server compact and spread rapidly
Sobig.F	Late 2003	exploited open proxy servers to turn infected machines into spam engines
Mydoom	2004	mass-mailing e-mail worm installed a backdoor in infected machines
Warezov	2006	creates executables in system directories sends itself as an e-mail attachment can disable security related products
Conficker (Downadup)	November 2008	exploits a Windows buffer overflow vulnerability most widespread infection since SQL Slammer
Stuxnet	2010	restricted rate of spread to reduce chance of detection targeted industrial control systems

Worm Technology



Mobile Code

- programs that can be shipped unchanged to a variety of platforms
- transmitted from a remote system to a local system and then executed on the local system
- often acts as a mechanism for a virus, worm, or Trojan horse
- takes advantage of vulnerabilities to perform it own exploits
- popular vehicles include Java applets, ActiveX, JavaScript and VBScript

Mobile Phone Worms

- first discovery was Cabir worm in 2004
- then Lasco and CommWarrior in 2005
- communicate through Bluetooth wireless connections or MMS
- target is the smartphone
- can completely disable the phone, delete data on the phone, or force the device to send costly messages
- CommWarrior replicates by means of Bluetooth to other phones, sends itself as an MMS file to contacts and as an auto reply to incoming text messages

Drive-By-Downloads

- exploits browser vulnerabilities to download and install malware on the system when the user views a Web page controlled by the attacker
- in most cases does not actively propagate
- spreads when users visit the malicious Web page

Social Engineering

- "tricking" users to assist in the compromise of their own systems

spam

- unsolicited bulk e-mail
- significant carrier of malware
- used for phishing attacks

Trojan horse

- program or utility containing harmful hidden code
- used to accomplish functions that the attacker could not accomplish directly

mobile phone trojans

- first appeared in 2004 (Skuller)
- target is the smartphone

Payload
System Corruption

- data destruction

  - Chernobyl virus
    - first seen in 1998
    - Windows 95 and 98 virus
    - infects executable files and corrupts the entire file system when a trigger date is reached

  - Klez
    - mass mailing worm infecting Windows 95 to XP systems
    - on trigger date causes files on the hard drive to become empty
  - ransomware
    - encrypts the user's data and demands payment in order to access the key needed to recover the information
    - PC Cyborg Trojan (1989)
    - Gpcode Trojan (2006)

Payload
System Corruption

- real-world damage
  - causes damage to physical equipment
    - Chernobyl virus rewrites BIOS code
  - Stuxnet worm
    - targets specific industrial control system software
  - there are concerns about using sophisticated targeted malware for industrial sabotage
- logic bomb
  - code embedded in the malware that is set to "explode" when certain conditions are met

Payload – Attack Agents
Bots

- takes 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 helper objects (BHOs)
  - attacking IRC chat networks
  - manipulating online polls/games

Remote Control Facility

- distinguishes a bot from a worm
  - worm propagates itself and activates itself
  - bot is initially controlled from some central facility
- typical means of implementing the remote control facility is on an IRC server
  - bots join a specific channel on this server and treat incoming messages as commands
  - more recent botnets use covert communication channels via protocols such as HTTP
  - distributed control mechanisms use peer-to-peer protocols to avoid a single point of failure

Payload – Information Theft Keyloggers and Spyware

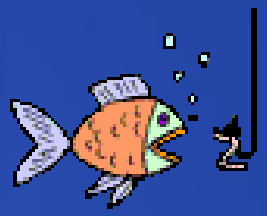
keylogger

- captures keystrokes to allow attacker to monitor sensitive information
- typically uses some form of filtering mechanism that only returns information close to keywords ("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

Payload – Information Theft Phishing

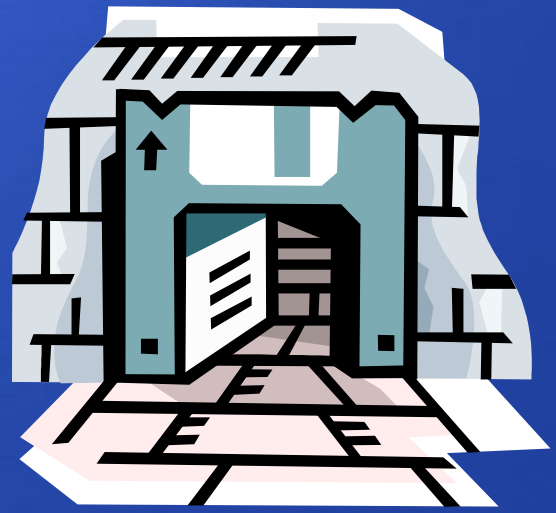


- exploits social engineering to leverage the user's trust by masquerading as communication from a trusted source
  - include a URL in a spam e-mail that links to a fake Web site that mimics the login page of a banking, gaming, or similar site
  - suggests that urgent action is required by the user to 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

Payload – Stealthing Backdoor

- also known as a *trapdoor*
- secret entry point into a program allowing the attacker to gain access and bypass the security access procedures
- *maintenance hook* is a backdoor used by programmers to debug and test programs
- difficult to implement operating system controls for backdoors in applications



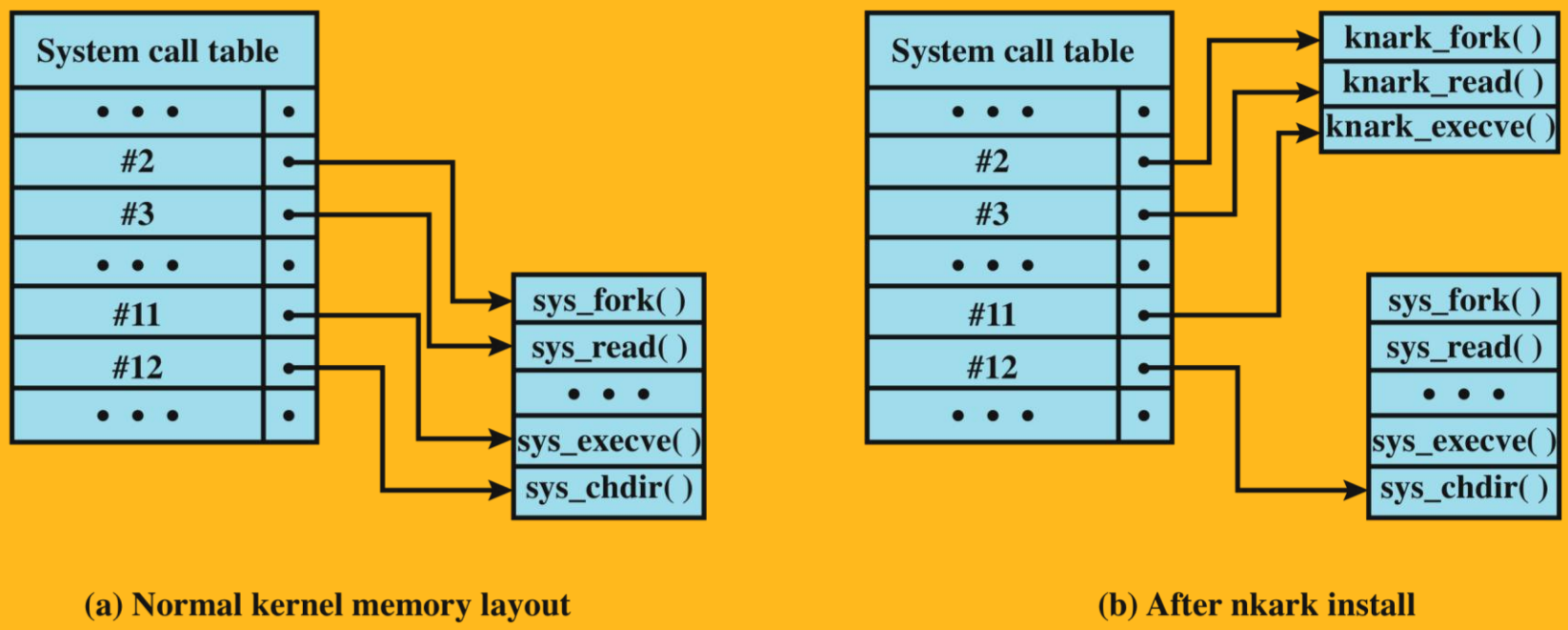
Payload - Stealthing Rootkit

- 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

Rootkit Classification Characteristics

persistent	memory based	user mode
kernel mode	virtual machine based	external mode

System Call Table Modification



(a) Normal kernel memory layout

(b) After nkark install

Figure 6.5 System Call Table Modification by Rootkit (based on [LEVI06])

Malware Countermeasure Approaches

- ideal solution to the threat of malware is prevention

four main elements of prevention:

- policy
- awareness
- vulnerability mitigation
- threat mitigation

- if prevention fails, technical mechanisms can be used to support the following threat mitigation options:
- detection
- identification
- removal

Generations of Anti-Virus Software

first generation: simple scanners

- requires a malware signature to identify the malware
- limited to the detection of known malware

second generation: heuristic scanners

- uses heuristic rules to search for probable malware instances
- another approach is integrity checking

third generation: activity traps

- memory-resident programs that identify malware by its actions rather than its structure in an infected program

fourth 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 (GD)

- enables the anti-virus program to easily detect complex polymorphic viruses and other malware while maintaining fast scanning speeds
- executable files are run through a GD scanner which contains the following elements:
  - CPU emulator
  - virus signature scanner
  - emulation control module
- the most difficult design issue with a GD scanner is to determine how long to run each interpretation

Host-Based Behavior-Blocking Software

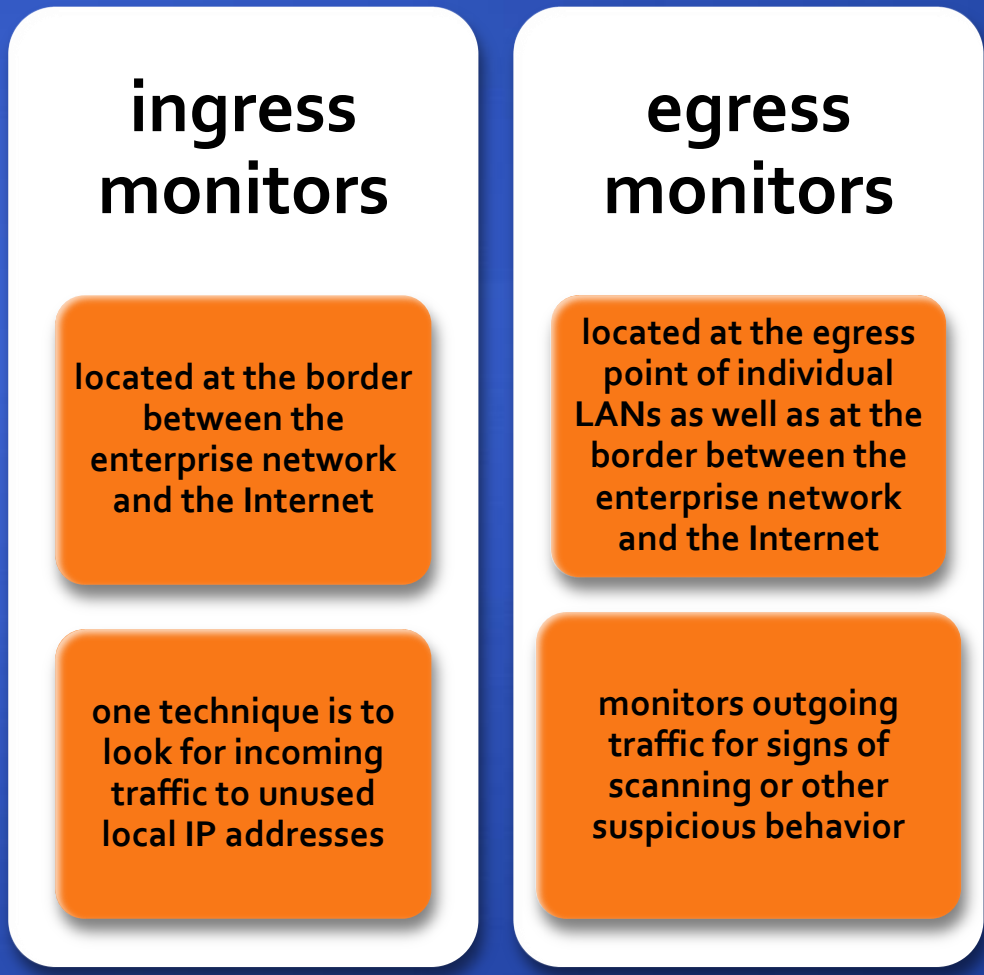
- integrates with the operating system of a host computer and monitors program behavior in real time for malicious action
  - blocks potentially malicious actions before they have a chance to affect the system
  - blocks software in real time so it has an advantage over anti-virus detection techniques such as fingerprinting or heuristics

limitations

- because malicious code must run on the target machine before all its behaviors can be identified, it can cause harm before it has been detected and blocked

Perimeter Scanning Approaches

- anti-virus software typically included in e-mail and Web proxy services running on an organization's firewall and IDS
- may also be included in the traffic analysis component of an IDS
- may include intrusion prevention measures, blocking the flow of any suspicious traffic
- approach is limited to scanning malware



two types of monitoring software

Worm Countermeasures

- considerable overlap in techniques for dealing with viruses and worms
- once a worm is resident on a machine anti-virus software can be used to detect and possibly remove it
- perimeter network activity and usage monitoring can form the basis of a worm defense
- worm defense approaches include:
  - signature-based worm scan filtering
  - filter-based worm containment
  - payload-classification-based worm containment
  - threshold random walk (TRW) scan detection
  - rate limiting
  - rate halting

Digital Immune System

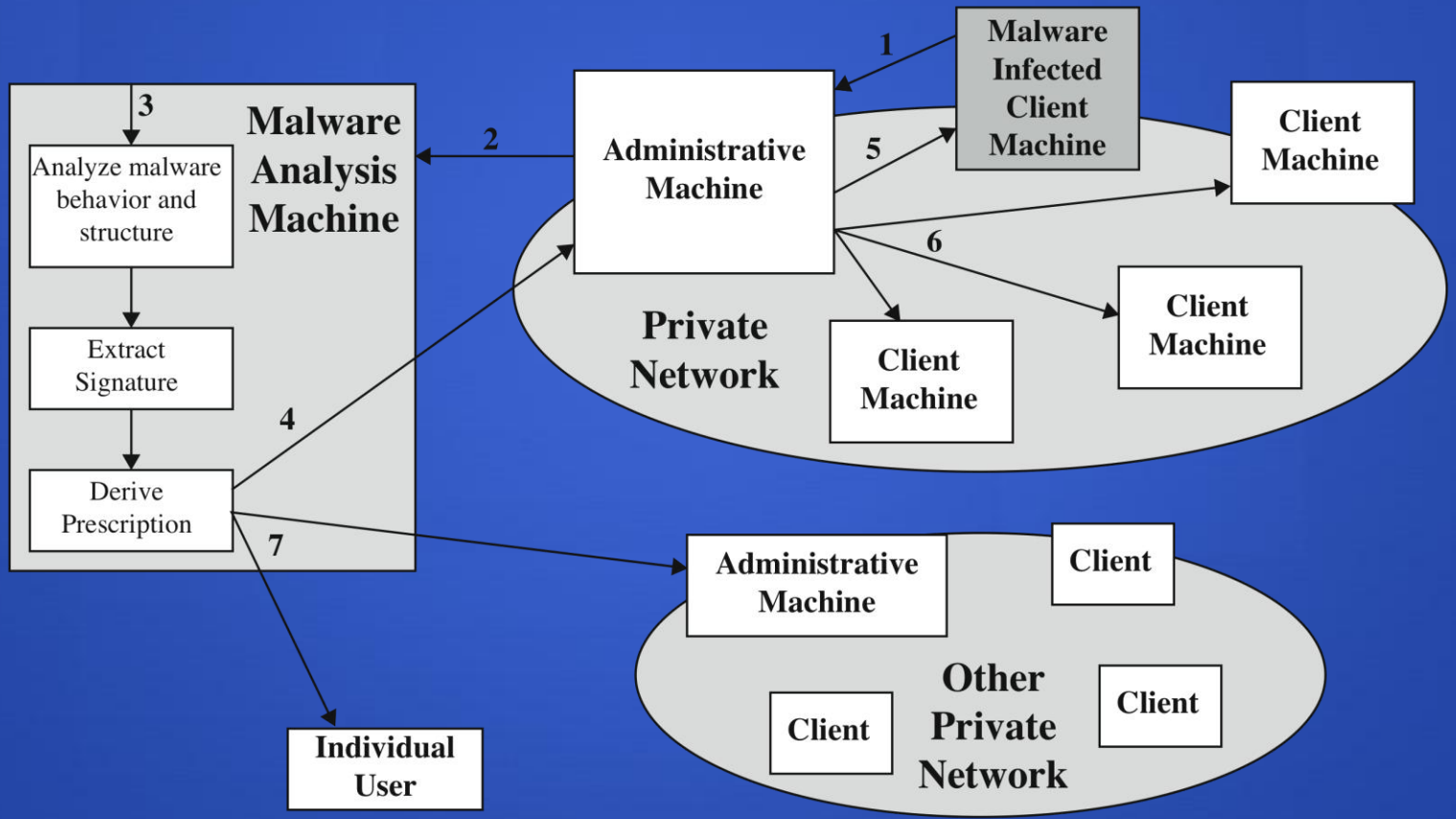


Figure 6.6 Digital Immune System

Worm Countermeasure Architecture

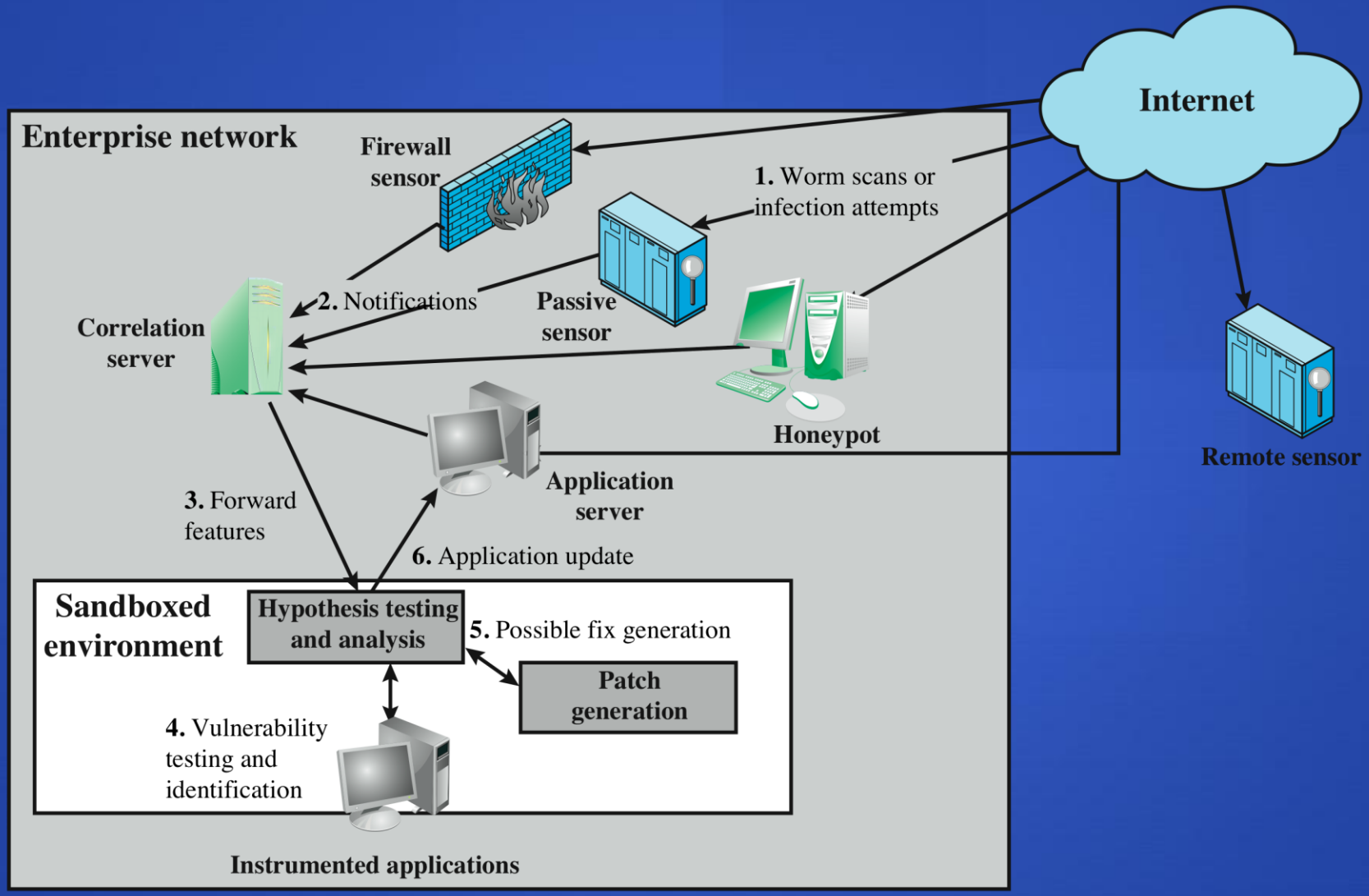


Figure 6.7 Placement of Worm Monitors (based on [SIDI05])

Summary

- types of malicious software (malware)
- terminology for malicious software
- viruses – infected content
  - infection mechanism, trigger, payload
  - dormant, propagation, triggering, and execution phases
  - boot sector infector, file infector, macro virus, and multipartite virus
  - encrypted, stealth, polymorphic, and metamorphic viruses
- worms – vulnerability exploit
  - replicates via remote systems
  - e-mail, file sharing, remote execution, remote file access, remote login capability
  - scanning/fingerprinting
- spam e-mail/trojans – social engineering
- payload – system corruption
  - data destruction, real world damage
  - ramsomware, logic bomb
- payload – attack agent
  - bots
  - remote control facility
- payload – information theft
  - credential theft, keyloggers, spyware
  - phishing, identity theft
- payload – stealthing
  - backdoor/trapdoor
  - rootkit
  - kernel mode rootkits
  - virtual machine/external rootkits
- countermeasures
  - prevention
  - detection, identification, removal
  - host based scanners/behavior blocking software
  - digital immune system