Fundamentals of Information & Network Security
ECE 471/571

Lecture #37: Malicious Software

Instructor: Ming Li

Dept of Electrical and Computer Engineering

University of Arizona

Malicious Software

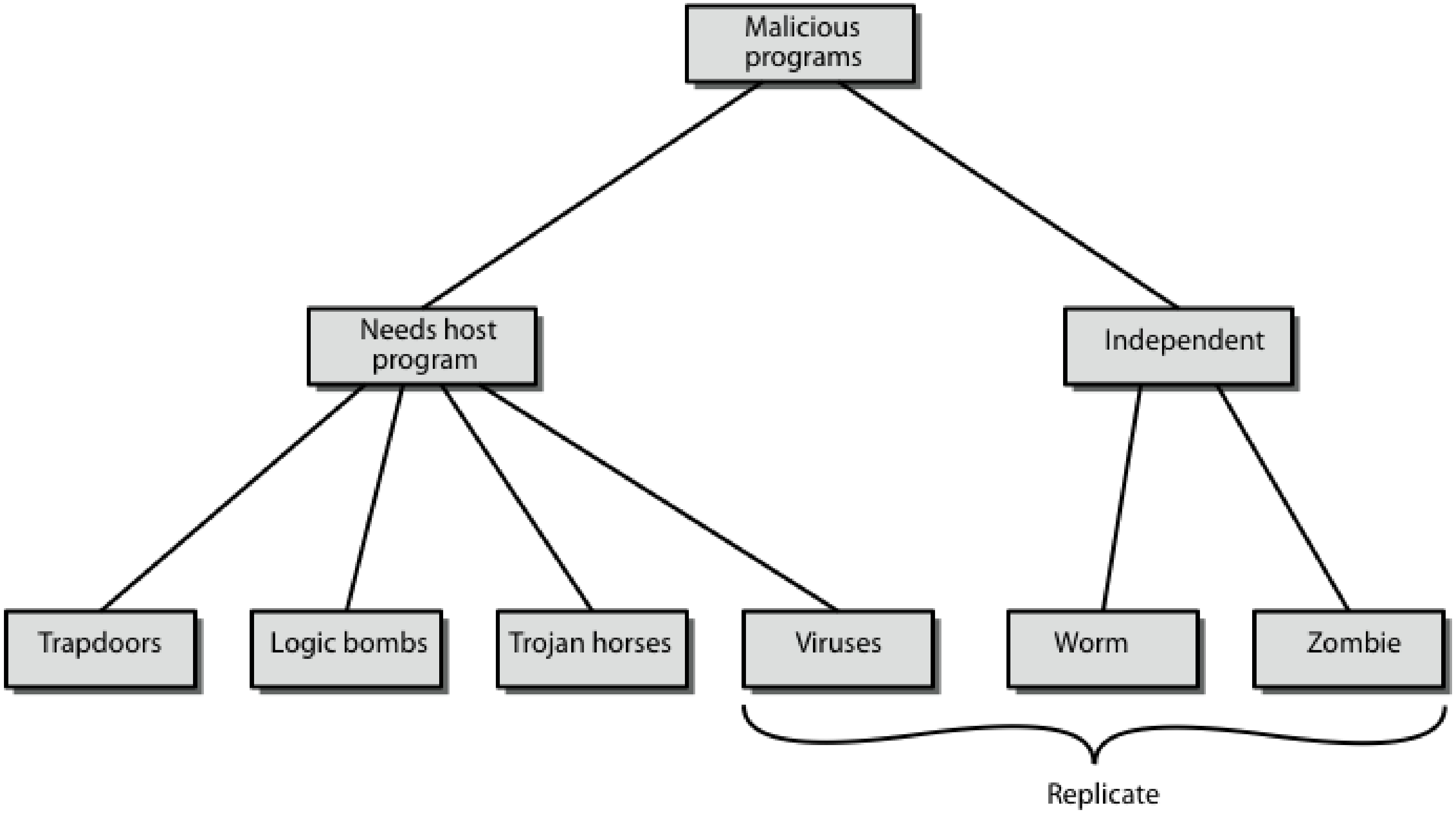
What is the concept of defense: The parrying of a blow. What is its characteristic feature: Awaiting the blow.

—*On War,* Carl Von Clausewitz

Viruses and Other Malicious Content

- computer viruses have got a lot of publicity
- one of a family of **malicious software**
- effects usually obvious
- have figured in news reports, fiction, movies (often exaggerated)
- getting more attention than deserve
- are a concern though

Malicious Software



Backdoor or Trapdoor

- secret entry point into a program
- allows those who know access bypassing usual security procedures
- have been commonly used by developers
- a threat when left in production programs allowing exploited by attackers
- very hard to block in O/S
- requires good s/w development & update

Logic Bomb

- one of oldest types of malicious software
- code embedded in legitimate program
- activated when specified conditions met
  - eg presence/absence of some file
  - particular date/time
  - particular user
- when triggered typically damage system
  - modify/delete files/disks, halt machine, etc

Trojan Horse

- program with hidden side-effects
- which is usually superficially attractive
  - eg game, s/w upgrade etc
- when run performs some additional tasks
  - allows attacker to indirectly gain access they do not have directly
- often used to propagate a virus/worm or install a backdoor
- or simply to destroy data

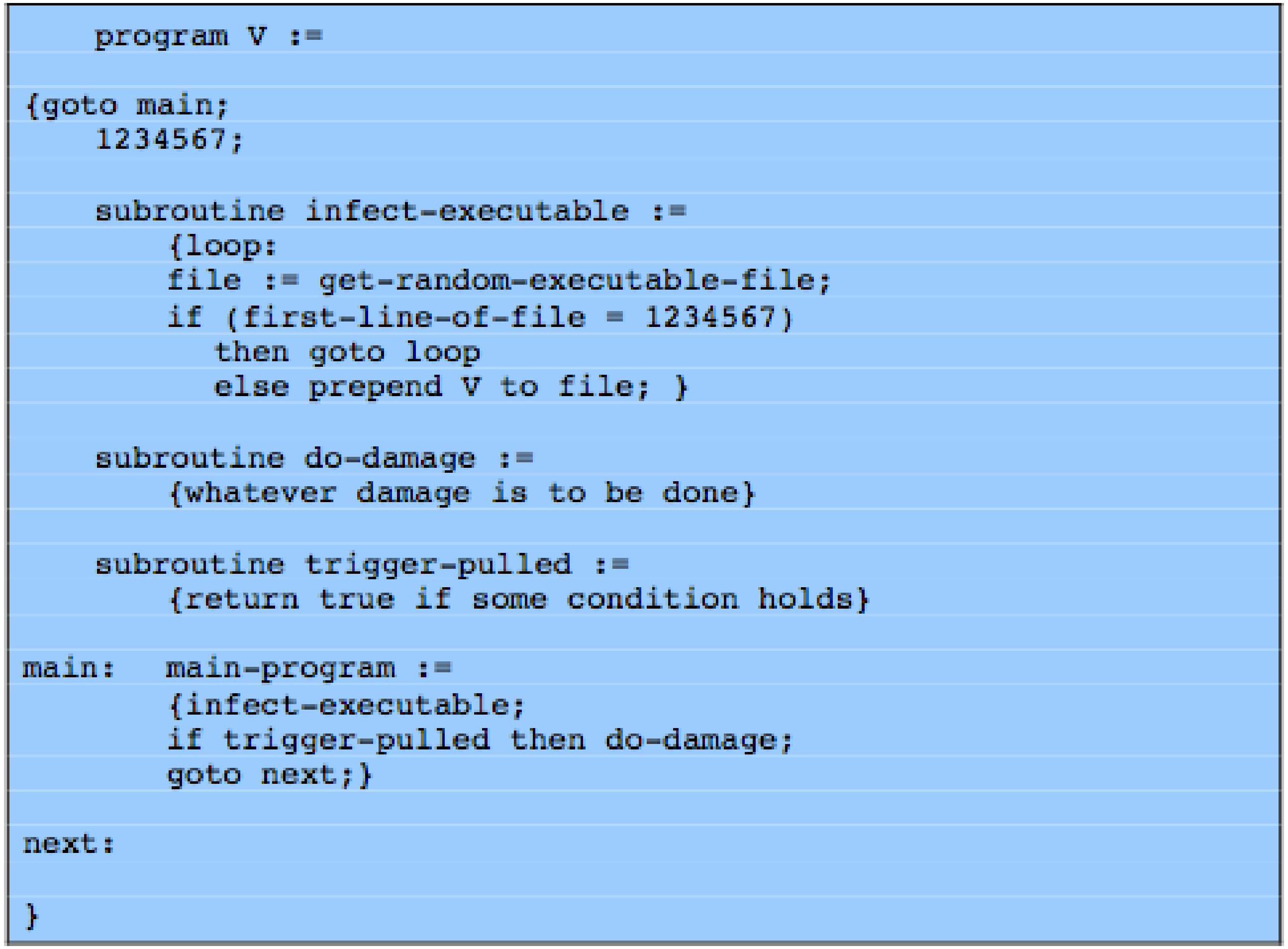
Viruses

- piece of software that infects programs
  - modifying them to include a copy of the virus
  - so it executes secretly when host program is run
- specific to operating system and hardware
  - taking advantage of their details and weaknesses
- a typical virus goes through phases of:
  - dormant
  - propagation
  - triggering
  - execution

Virus Structure

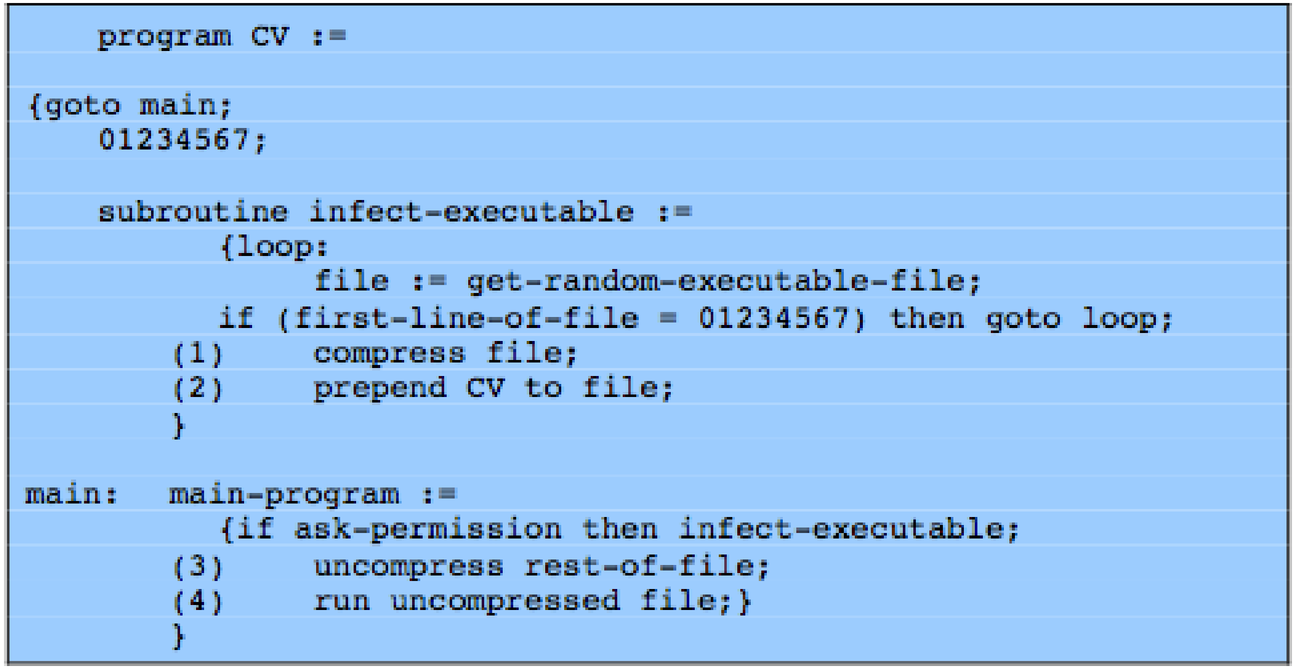
- components:
  - infection mechanism - enables replication
  - trigger - event that makes payload activate
  - payload - what it does, malicious or benign
- prepended / postpended / embedded
- when infected program invoked, executes virus code then original program code
- can block initial infection (difficult)
- or propogation (with access controls)

Virus Structure

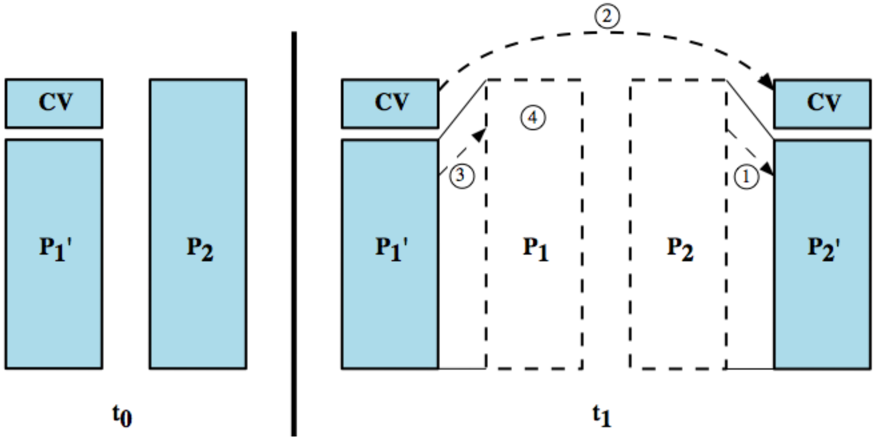


```
    program V :=

{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:

}
```

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 Classification

- boot sector
- file infector
- macro virus

- encrypted virus
- stealth virus
- polymorphic virus
- metamorphic virus

Virus Countermeasures

- prevention - ideal solution but difficult
- realistically need:
  - detection
  - identification
  - removal
- if detect but can’t identify or remove, must discard and replace infected program

Exercise #4

The point of this problem is to demonstrate the type of puzzles that must be solved in the design of malicious code and therefore, the type of mindset that one wishing to counter such attacks must adopt.

a. Consider the following C program:

```
begin
    print (*begin print (); end.*);
end
```

What do you think the program was intended to do? Does it work?

b. What is the specific relevance of this problem to this chapter?

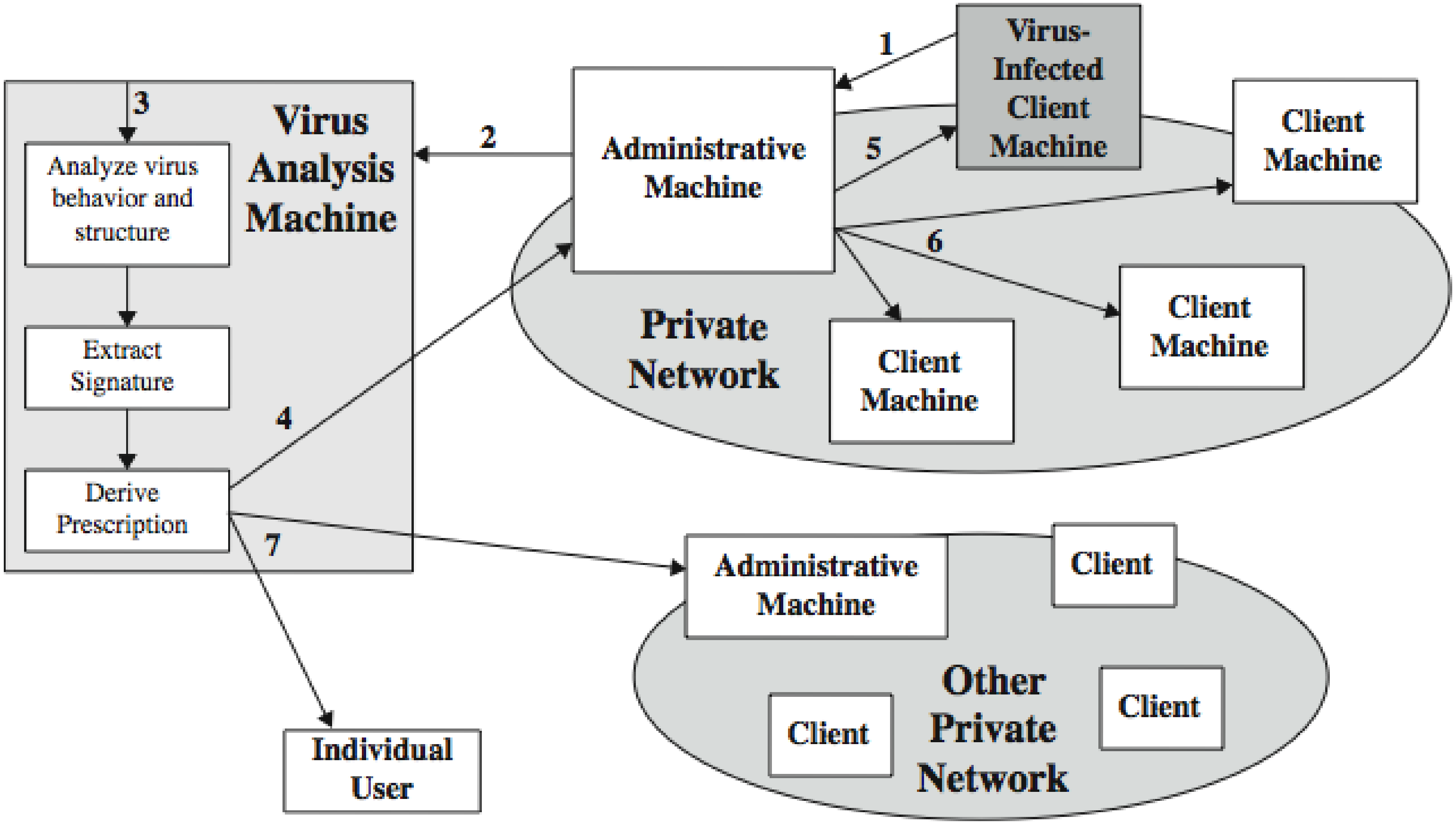
Anti-Virus Evolution

- virus & antivirus tech have both evolved
- early viruses simple code, easily removed
- as become more complex, so must the countermeasures
- generations
  - first - signature scanners
  - second - heuristics
  - third - identify actions
  - fourth - combination packages

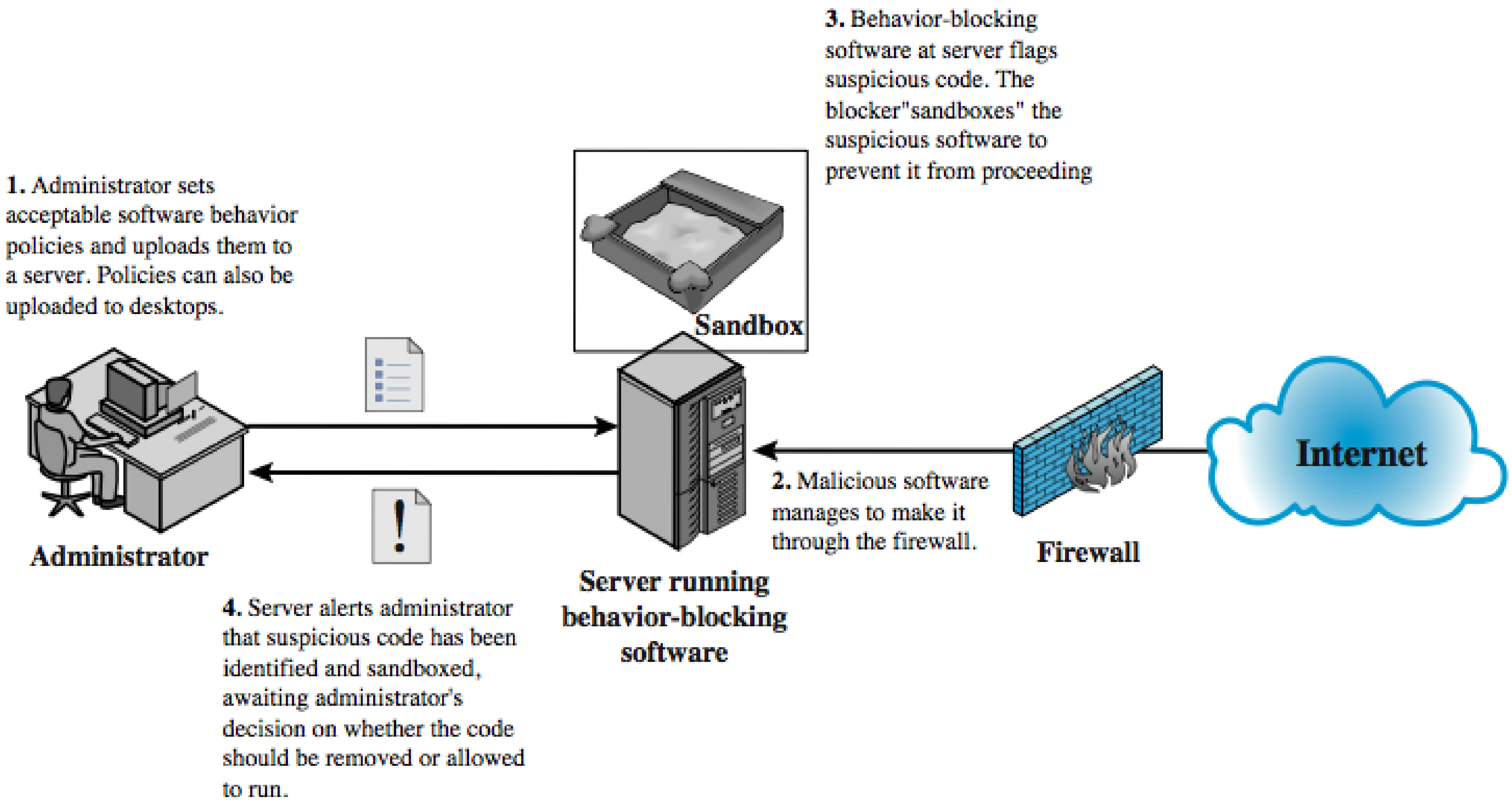
Generic Decryption

- runs executable files through GD scanner:
  - CPU emulator to interpret instructions
  - virus scanner to check known virus signatures
  - emulation control module to manage process
- let virus decrypt itself in interpreter
- periodically scan for virus signatures
- issue is long to interpret and scan
  - tradeoff chance of detection vs time delay

Digital Immune System



Behavior-Blocking Software



1. Administrator sets acceptable software behavior policies and uploads them to a server. Policies can also be uploaded to desktops.

2. Malicious software manages to make it through the firewall.

3. Behavior-blocking software at server flags suspicious code. The blocker"sandboxes" the suspicious software to prevent it from proceeding

4. Server alerts administrator that suspicious code has been identified and sandboxed, awaiting administrator's decision on whether the code should be removed or allowed to run.

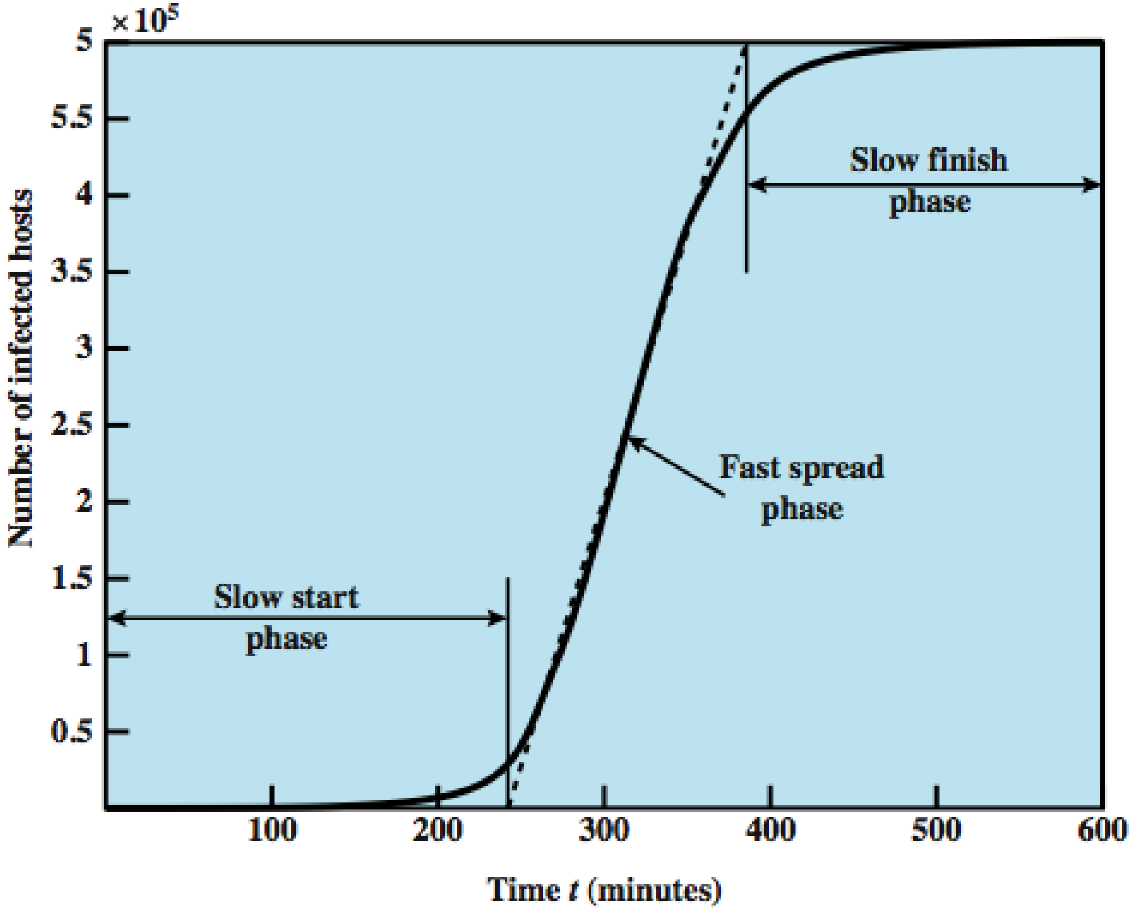
Worms

- replicating program that propagates over net
  - using email, remote exec, remote login
- has phases like a virus:
  - dormant, propagation, triggering, execution
  - propagation phase: searches for other systems, connects to it, copies self to it and runs
- may disguise itself as a system process
- concept seen in Brunner's "Shockwave Rider"
- implemented by Xerox Palo Alto labs in 1980's

Morris Worm

- one of best know worms
- released by Robert Morris in 1988
- various attacks on UNIX systems
  - cracking password file to use login/password to logon to other systems
  - exploiting a bug in the finger protocol
  - exploiting a bug in sendmail
- if succeed have remote shell access
  - sent bootstrap program to copy worm over

Worm Propagation Model



Recent Worm Attacks

- Code Red
  - July 2001 exploiting MS IIS bug
  - probes random IP address, does DDoS attack
- Code Red II variant includes backdoor
- SQL Slammer
  - early 2003, attacks MS SQL Server
- Mydoom
  - mass-mailing e-mail worm that appeared in 2004
  - installed remote access backdoor in infected systems
- Warezov family of worms
  - scan for e-mail addresses, send in attachment

Summary

- have considered:
  - various malicious programs
  - trapdoor, logic bomb, trojan horse, zombie
  - viruses
  - worms