Malicious Logic

Week 8, Tuesday

Introduction

• Malicious logic: a set of instructions that cause a site’s security policy to be violated
• They make most traditional access control useless
  – Typically by assuming an authorized user’s identity
• We will talk Trojan horses, viruses, worms, etc.

Trojan Horse

• An example:
  A Unix script called “ls” in directory /foo, created by EVE:
  ```
  cp /bin/sh /tmp/.xxsh
  chmod o+s,a+x /tmp/.xxsh
  rm ./ls
  ls $*
  ```
  If user VICTOR is cheated to run the above “ls” instead of the original “ls” he will create a /tmp/.xxsh file. EVE then can run /tmp/.xxsh, a shell program that allows EVE to enjoy a shell running environment for VICTOR!
• This “ls” by EVE does do what the original “ls” does
  – But it also does something else: creating a shell program that can give EVE the access rights of VICTOR!

Computer Viruses

• Computer virus: a program that inserts itself into one or more programs and then performs some action
  – A Trojan horse only propagates itself
  – A virus infects others

• Note: some experts regard virus as a type of Trojan horse, where the infected program gives the overt effect and the virus code leads to covert effect.

Virus Mechanism

beginvirus:
  if spread-condition then begin
    for some set of target files do begin
      if target is not infected then begin
        1. determine where to place virus instructions
        2. copy virus instructions from beginvirus to endvirus into target
        3. alter target to execute added instructions
      end
    end
  end
  perform some action(s)
  goto beginning of infected program
endvirus
There are many many viruses . . .

• And there will be many many many of them!
• Types:
  – Boot sector infectors
  – Executable infectors
  – Multipartite viruses
  – TSR viruses
  – Stealth viruses
  – Encrypted viruses
  – Polymorphic viruses
  – Macro viruses

Boot Sector Infectors

• Boot sector: the part of a disk used to bootstrap the system
  – Or mount a disk
• Boot sector is executed when the system “sees” the disk for the first time
• **Boot sector infectors**: a virus that inserts itself into the boot sector of a disk
  – Loaded into memory when invoked

Executable Infectors

• **Executable infectors**: a virus that infects executable programs (or applications)
  – Also called COM or EXE viruses
  – The virus prepends or appends itself

Multipartite Viruses

• **Multipartite virus**: a virus that can infect either boot sectors or applications
  – Often has two parts, one for each type
    • Depending on what to infect

TSR Viruses

• **Terminate and stay resident (TSR) virus**: a virus that stays active (resident) in memory after the application (or bootstrapping, or disk mounting) has terminated
  – Can be boot sector infectors or executable infectors

Stealth Viruses

• **Stealth virus**: a virus that conceal the infection of files.
• For example:
  – If you try to read an infected program to check the integrity of the program, the virus will return you the original program
  – But, the virus will still get executed when the program is executed
Encrypted Viruses

- **Encrypted virus**: a virus that uses enciphered code
  - In order to prevent virus detection

Macro Viruses

- **Macro virus**: a virus that composed of a sequence of instructions that is interpreted, rather than executed directly
- Can infect either executable applications, or data files
  - The latter called data virus

Polymorphic Viruses

- **Polymorphic virus**: a virus that changes its form each time it inserts itself into another program
  - For example, using different machine instructions with the same effects
  - Can be automated
  - An encrypted virus can evolve into a polymorphic virus by varying the decipherment routine

Computer Worms

- **Computer worm**: a program that copies from one computer to another
  - A variant of virus
- Many worms have occurred
  - Father Christmas worm, CodeRed worm, Slammer worm, etc.
  - The Slammer worm spread worldwide in ~5 minutes

Rabbits and Bacteria

- **Rabbit or bacterium**: a program that absorbs all or some class of resource
  - Creating denial-of-service attack
- For example
  ```
  while (true) 
    do
      mkdir x
      cd x
  done
  ```

Logic Bomb

- **Logic bomb**: a program that performs a malicious action when some external event occurs
Theory of Malicious Logic

• Theorem 22-1: It is undecidable whether an arbitrary program contains a computer virus
  – Proof: omitted
• Theorem 22-1: It is undecidable whether an arbitrary program contains a malicious logic
  – Proof: omitted

Defenses

• Malicious logic acting as both data & instructions
• Malicious logic assuming the identity of a user
• Malicious logic crossing protection domain boundaries by sharing
• Malicious logic altering files
• Malicious logic performing actions beyond specification
• Malicious logic altering statistical characteristics

Malicious Logic Acting as Both Data and Instructions

• When malicious code is inserted, it is data
• When malicious code is executed, it is instructions
• Solution
  – Treat all programs as “data”
  – Must be certified to become “executables”
    • Certification process must be secure
  – Modifying an “executable” will make it become “data” again

Malicious Logic Assuming the Identity of a User

• Limit the distance a virus can spread
  – Flow distance metric \( f_d(x) \) : info \( x \) can only travel for distance \( f_d(x) \).
• Reduce the rights
  – Principle of least privilege
  – Authorization
  – “watchdog” or “guardian” to check each access
• Use sandbox or virtual machines
  – To restrict process rights in a confined environment

Malicious Logic Crossing Protection Domain Boundaries by Sharing

• Inhibit users from different domains from sharing programs or data
  – But then the sharing is also disabled
  – Thus many service will be disabled as well
• Hard!

Malicious Logic Altering Files

• Use signature block to sign a program
  – We have learned this!
• Must ensure no malicious code before the signing operation
Malicious Logic Performing Actions Beyond Specification

- Treat infection and execution of viruses as **error**!
- Example 1:
  - Break a program into sequences of nonbranching instructions
  - Sign each of them (and store the signatures)
- Example 2: N-version programming
  - N versions of a program is made
  - Run concurrently
  - Hopefully the majority of them will be correct

(cont’d)

- **Proof-carrying code (PCC):**
  - Code user specifies a safety requirement
  - Code author generates a proof that the code meets such requirement
  - And integrate the proof with the code!
  - The code user then verifies the proof

Malicious Logic Altering Statistical Characteristics

- Malicious code may change the specific statistical characteristics that a program originally had
  - So the detection of such changes can help detect the attack
- Example:
  - The printing command “lpr” is used 1000 times by Joe in just one day

Midterm Summary

CIS 410 (Undergrads):
Average: 63.69
Highest: 83.5

CIS 510 (Grads):
Average: ~79
Highest: 94

Class Scheduling Changes

- On week 9, Tuesday, May 27
  *Class is cancelled*
- On week 9, Thursday, May 29
  *Jin Zhang will review midterm & homework*