Malicious Software

CSS 322 – Security and Cryptography

Contents

- Terminology and Classification
- Viruses
- Worms

Classifying Malicious Programs

- Host Dependence
  - Host Dependent: Code/programs are embedded in actual programs, e.g. viruses, backdoors
  - Host Independent: Programs can be run separately by OS, e.g. worms, zombies
- Replication
  - Non-replicating: programs usually activated by a trigger, e.g. logic bombs, backdoors
  - Replicating: make copies of themselves, e.g. viruses, worms

Terminology of Malicious Programs

- **Virus**: Attaches itself to a program and propagates copies of itself to other programs
- **Worm**: Program that propagates copies of itself to other computers
- **Logic bomb**: Triggers action when condition occurs
- **Trojan horse**: Program that contains unexpected additional functionality
- **Backdoor** (**trapdoor**): Program modification that allows unauthorized access to functionality
- **Exploits**: Code specific to a single vulnerability or set of vulnerabilities
- **Downloaders**: Program that installs other items on a machine that is under attack. Usually, a downloader is sent in an e-mail.
- **Auto-rooter**: Malicious hacker tools used to break into new machines remotely
- **Kit** (virus generator): Set of tools for generating new viruses automatically
- **Spammer programs**: Used to send large volumes of unwanted e-mail
- **Flooders**: Used to attack networked computer systems with a large volume of traffic to carry out a denial of service (DoS) attack
- **Keyloggers**: Captures keystrokes on a compromised system
- **Rootkit**: Set of hacker tools used after attacker has broken into a computer system and gained root-level access
- **Zombie Program**: activated on an infected machine that is activated to launch attacks on other machines

Backdoor

- Secret entry point into a program to allow attacker to gain access, bypassing normal security access control
- Programmers use backdoors for legitimate testing procedures
  - When testing or debugging, often a programmer will want to avoid going through authentication procedures, or lengthy logins
  - Programmer issue a special set of commands that bypass normal procedures (e.g. special user ID or sequence of inputs)
- Backdoors are malicious when programmers create and use backdoors to gain unauthorised access to real systems

Logic Bomb

- Code embedded in a program that executes when certain conditions are met:
  - Absence or presence of certain files
  - Date or time
  - Particular user executing a command
- Once triggered, the bomb may perform malicious operations:
  - Delete, modify files
  - Crash a computer
  - Send information to another computer
- Example: ex-employees leave logic bombs in company; Tim Lloyd was chief network engineer and 20 days after fired a logic bomb deleted most of the company software design and code; cost more than $US10m; Lloyd was jailed for 3 years

Nature of Viruses

- A virus is piece of software that “infects” programs and copies itself to other programs
- The phases of a virus are:
  - Dormant: virus is idle; will be activated by some event (like logic bomb)
  - Propagation: virus copies itself into other programs or areas of operating system
  - Triggering: virus is activated to perform some function; similar triggers to logic bombs, but also number of times virus copied
  - 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

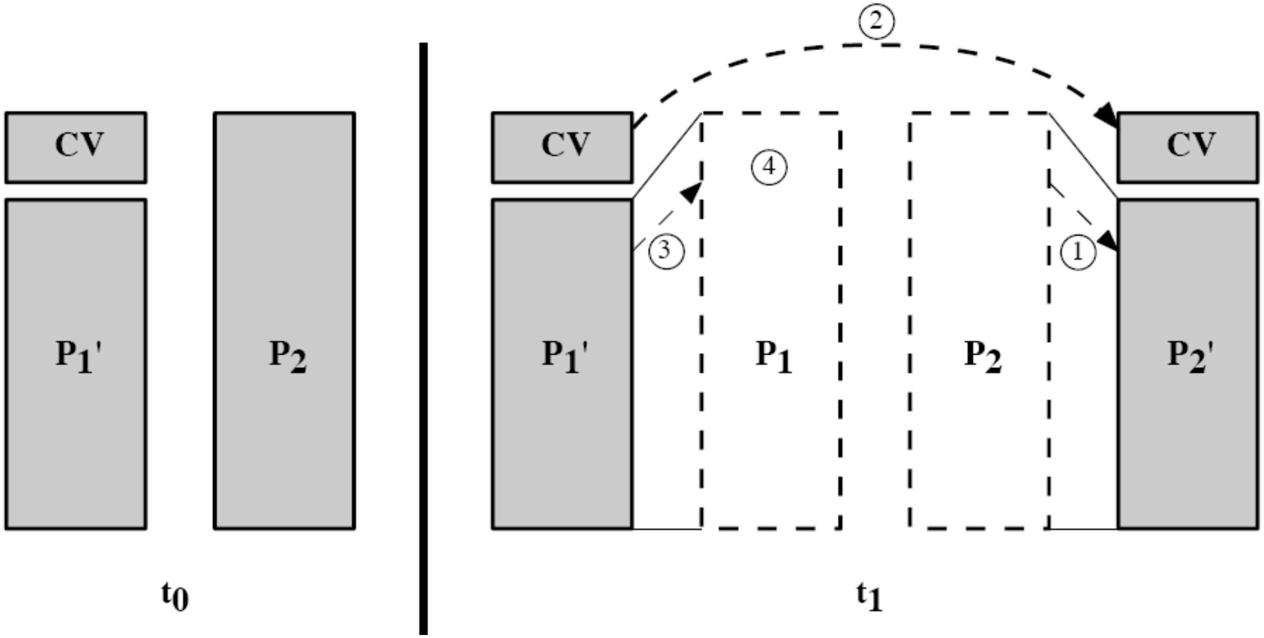
```
program V :=

{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



Compression Virus Algorithm

```
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

- *Parasitic Virus*: virus attaches to executable file and copies itself to other executables that it can find
- *Memory-resident virus*: stored in main memory as part of current program executing; infects other programs that execute
- *Boot sector virus*: stored in boot sector of hard or floppy disk; spreads when system boots from disk (a popular method before computer networks were widespread)
- *Polymorphic virus*: changes (mutates) with each copy, so harder to detect based on signatures
  - E.g. Add extra, redundant code; re-order code
- *Metamorphic virus*: change appearance as well as behaviour
  - Very hard to detect

Macro Viruses

- Macro viruses became most common type of virus in 1990’s
- Reasons for threat of macro viruses:
  - Most macro viruses are for Microsoft based applications (e.g. Word, Excel) which are very common; can infect any computer system that uses these applications
  - Infect documents, not programs; documents are more widespread (and exchanged much more often) than executable programs; users are (were?) less suspecting of documents than executables
- Macros are executable programs embedded in documents

Email Viruses

- Macro viruses and viruses in executables require the user to run the program (e.g. open the Word document)
  - These mainly are sent by email
- Visual Basic scripting capabilities of email clients (e.g. Microsoft Outlook) allowed viruses to be written and run by just opening an email (not the attachment)
  - Much easier to spread and harder to prevent users from opening
  - Requires safe use of Internet utilities and applications (e.g. safe scripting languages, or no scripting)

Distribution of Viruses/Worms

- Assume a worm infects 4 new computer every hour

time in hours	number of new victims
1	4
2	16
3	64
4	256
5	1024
6	4096
7	16384
8	65536
9	262144
10	1048576
24	10^{14}

Only 10^{10} people in world!

Melissa Virus

- Virus released by David Smith in 26 March 1999
  - Posted a message to a newsgroup containing a MS Word attachment – the attachment contained a macro virus
  - Estimated damage up to $US1000million (34 billion Baht)
    - Mainly cost of downtime (users not working) and removing virus from systems
  - Smith was arrested in 1 April 1999
    - After deals, Smith spent about 2 years in prison
- Designed to infect computers with Word 97/2000
  - Virus sent as attachment to email
    - Subject: “Important message from <username>”
    - Body: “Here is that document you asked for … don’t show anyone else”
  - When executed, the macro automatically sent the email to 50 people in address book
    - Required MS Outlook to be running
    - Look like you receive an email from someone you know
  - Macro would also copy itself into normal.dot (the standard template for Word) – therefore infect all other documents created on the computer

Worms

- Software that replicates itself and sends copies to other computers
  - And copies on new computers repeat the process (copy and send)
  - May perform some function as well (e.g. delete files)
- Is an email virus a virus or worm or both?
  - Email virus requires users to propagate
  - Worms propagate by themselves (without user intervention)
- Worms use network connections to propagate:
  - Email software, e.g. Simple Mail Transfer Protocol (SMTP)
  - Remote execution, Remote Procedure Call, sockets
  - Remote login, e.g. telnet, rlogin, rsh, …
- Three main steps of worm:
  1. Search for other systems to infect
  2. Connect to a remote system
  3. Copy itself to remote system and cause the copy to execute

Morris Worm

- Robert Morris (undergrad at Cornell) released worm on Internet in 1988
  - One of the first major worms on the Internet
  - Infected about 3000 computers; 5% of the Internet
  - Caused shutdown of Internet for several days
    - Cost of repair between $US100,000 and $US10,000,000
  - Morris was one of first people arrested, tried and convicted for releasing malicious computer program
    - Received 3 years probation (no prison time) and $US10,000 fine
- Spread on UNIX systems (the main computers on the Internet at the time)
  - Worm propagated using UNIX remote login commands
  - Gain unauthorised access to systems using:
    - Legitimate trusted host features of rsh, rexec commands for remote login
    - Crack passwords using 432 common passwords, variations on username and a UNIX dictionary
    - Exploit a bug in sendmail
    - Exploit a buffer overflow bug in fingerd

Code Red

- CodeRed (16 July 2001)
  - Worm aimed at Microsoft Internet Information Server (IIS) web servers (not users)
  - Sent to web server as HTTP GET request
    - Bug in IIS allows the code to be stored by the server
    - Worm was stored in RAM; a reboot deleted the worm (but many web servers run 24 hours per day)
  - Worm had several states:
    - On first 19 days of month, send HTTP GET requests to random IP addresses, with the intention of infecting other web servers
    - On days 20 to 28 create a denial-of-service attack on www.whitehouse.gov
    - Dormant for remainder of month
  - Infected 200,000 servers in 5 hours
  - Consumed significant network resources (denial of service attack)
- CodeRed II (4 August 2001)
  - Similar to CodeRed but also installed a trojan horse on the web server
    - Allowed anyone with web browser to send commands to web server:
      - E.g. delete or modify files on server

I Love You Worm

- Reported on 4 May 2000; writers from Philippines
  - Damages up to $US9 billion
    - Infected more than half of US companies; 10,000 mail servers in Europe
  - 1 in 28 emails sent on Internet were from ILOVEYOU worm
  - Writers were identified but not arrested as was not a crime in Philippines
- Used similar mechanism as Melissa to propagate (except sent email to everyone in address book)
  - Not technically a virus: Did not infect other programs
  - Email included attachment: LOVE-LETTER-FOR-YOU.txt.vbs
  - When opened, executed a Visual Basic Script
    - Delete files from hard drive by replacing the file with the worm
    - Point web browser to site in Philippines to download a Trojan horse that collected passwords from victims machine and emailed them back to attacker

Current Trends in Worms

- New worms have new technologies:
  - Multiplatform: not limited to Windows, also Linux distributions and MAC
  - Mutliexploit: Exploit different bugs in web servers, client applications, P2P network software, email servers, …
  - Ultrafast spreading: utilise network software to first determine which computers have bugs (instead of randomly send to computers)
  - Polymorphic: avoid detection by create different copies that perform the same (look different but behave the same)
  - Metamorphic: avoid detection by creating copies that modify their behaviour
  - Zero-day exploit: Exploit vulnerabilities (bugs) that are unknown until the worm is released