7. Viruses

7.1 Introduction

7.2 Simple infections, logic bombs and trojan horses

7.3 Viruses

7.4 Worms

7.5 Antiviral fight/protection

7.6 Spyware

7.1. Introduction

- virology, based on artificial intelligence (mathematics + computer science)
- Anti-virus = non perfect product
- What is a virus? A program able to reproduce itself and to be propagated

7.1. Theoretical aspects

- Türing Machine
  - Abstract and general representation of a computer and programs likely to be carried out on this computer
  - Objective of these theoretical aspects: is a function F calculable? (in other words can we find an algorithm able to calculate it)
  - Concerning viruses, the question will be “is the function F (= self-reproduction) calculable?”
- Researches from Von Neumann
  - Cellular automats => proved that we could conceive self-reproductive programs

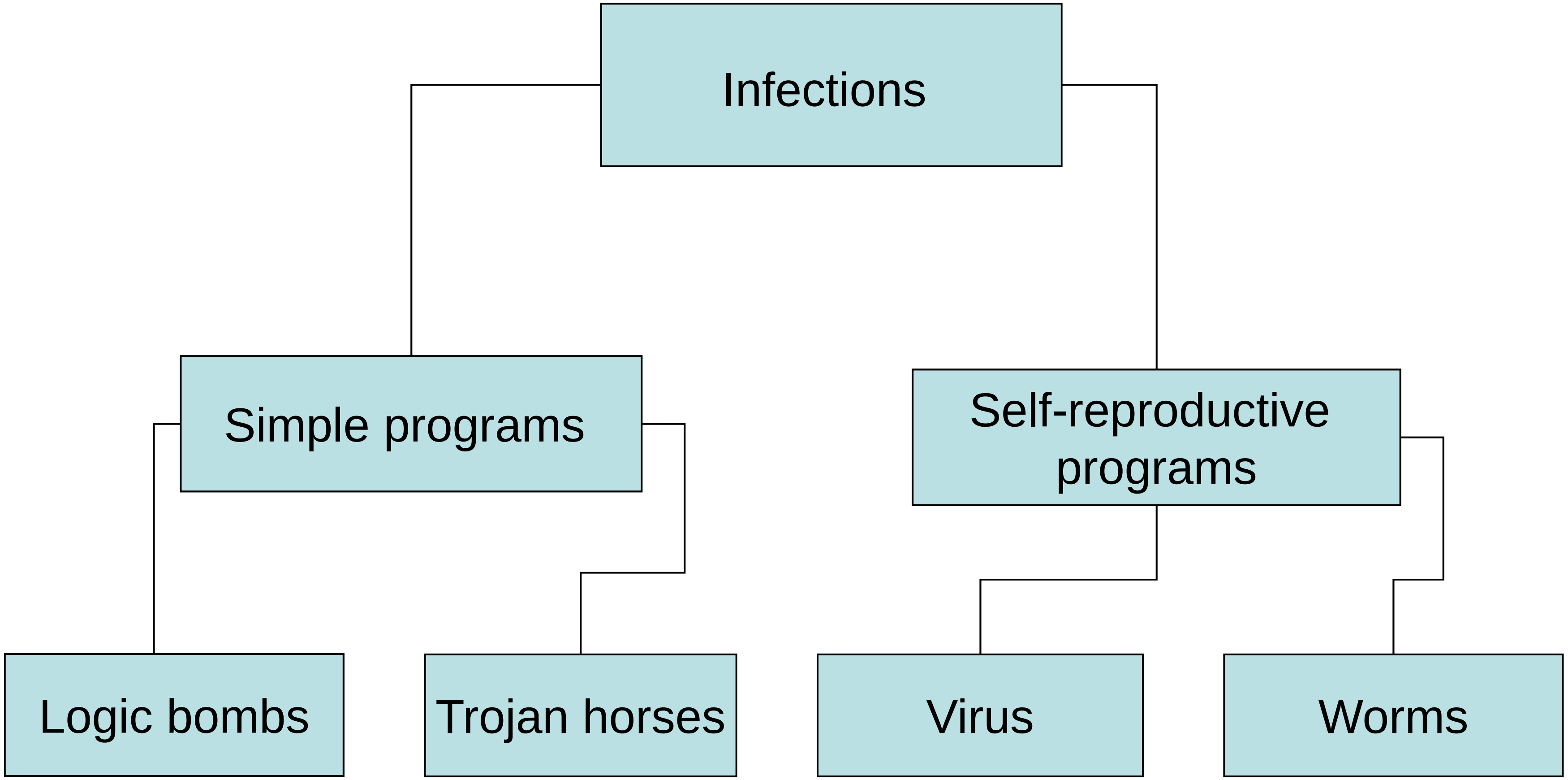
7.1. Historical aspects

- First known viruses: 1970
- In fact, American Defense (Arpanet)/MIT worked before (offensive and defensive code)
- Xerox Worm (1981)
- Elk Cloner Virus on DOS 3.3 (1983)
- Brain boot Virus (at the beginning of 80s)
- Thesis (Ph.D.) of Fred Cohen (univ. from Southern-California, 1986) defines the term and the concept of virus (defined by Cohen & Adleman, 1986)
  - "A virus is a sequence of symbols which, interpreted in a given environment (adequate), **modifies** other sequences of symbols in this environment, so as to **include a copy of itself** there, this copy **having possibly evolved/moved**"

7.1. Some figures

- 1999, CIH virus (known as Chernobyl), obliged thousands of users to change the central card of their computer after having destroyed BIOS card
- 1999, ILoveYou worm, more than 45.000.000 infected computers
- 2002, Sapphire/Slammer worm, + of 75.000 servers in approximately 10 minutes
- 2003, W32/Sobig.F worm, + of 100.000.000 users

7.1 Classification of the infections (*malware*)



7.2. Simple infections

- The purpose of these programs is simply to settle in the system:
  - Resident mode : active process in memory in a permanent way in order to be able to act as long as the system functions
  - Furtive mode: the user should not realize that such a program, resident, is present in its system (invisible with `ps –aux` in Unix or in the task manager of Windows)
  - Persistent mode: infecting program able to reinstall itself after removal or de-installation (for example by means of keys added in the register base)

7.2 Logic bombs

- Def.: simple infecting program, settling in the system and which awaits an event (particular date, action, data) called in general “trigger”, to carry out its offensive function
  - Ex: CIH 1.2 starts each 26 April
  - Ex: an administrator had established a program checking the presence of his name in the registers of payroll of his company. In the event of absence of this name (what means that the administrator was returned), the program encrypted all the hard disks…
- The anti-viruses have difficulties to detect logic bombs (before the update of the signature, in which case detection is systematic)

7.2 Trojan horses

- Def.: a Trojan is a simple program, composed of two parts, the server module and the client module. The server module, installed in the computer of the victim, gives discreetly to the attacker access to whole or part of its resources, which lays out about it via the network (in general) thanks to a client module (he is the “client” of the “services” delivered unconsciously by the victim)

7.2 Trojan horses

- The server module is dissimulated in an attractive program. The running of this program installs without the knowledge of the victim the server part of the Trojan
- The client module, once installed on the machine of the attacker, seeks on the network (order ping) the machines infected by the server module (IP addresses and TCP or UDP port)
  - Takeover allowing to carry out a more or less large number of offensive actions
    - Restarting the computer
    - File transfer
    - Execution of code
    - Destruction of data
    - Listen to keyboard
  - Ex: Back Orifice, Netbus, SubSeven
- To protect ourselves from trojan horses: firewall and antivirus (it is always possible to program a Trojan being able to pass through these protection tools…)

7.2 Port and protocol used by some trojan horses

Port	Protocol	Trojan
1024	TCP	NetSpy
1243	TCP	SubSeven
1999	TCP	Backdoor
6711	TCP	SubSeven
6712	TCP	SubSeven
6713	TCP	SubSeven
6776	TCP	SubSeven
12345	TCP	Netbus
12346	TCP	Netbus
12456	TCP	Netbus
20034	TCP	Netbus 2 Pro
31337	UDP	Back Orifice
54320	UDP	Back Orifice
54320	TCP	Back Orifice 2000

7.3. Functional diagram of a self-reproductive program (virus or worms)

- General structure
  - research routine of target programs
    - check that the target can be executed
    - check that the target is not already infected (often the viruses have a signature => this one is also detected by the anti-virus ones)
  - copy routine
    - copy in the target a copy of its own code
  - anti-detection routine
    - To be hidden from the anti-virus to ensure the survival of the virus
  - possibly a final load (optional destroying will), coupled or not with a differed trip
- Difference between self-reproductive programs…
  - code Duplication
- …and simple infection
  - No duplication

7.3 Life-phases of a virus (1/3)

- Infection phase
  - Passive
    - Dropper (program carrying a virus) copied from a support (CD, remote loading, forum) and transmitted
      - *Virus_1099* transmitted via pre-formatted virgin diskettes
      - *Warrier* diffused via a downloadable Packman game
      - *CIH* Virus in an official Yamaha driver or for IBM/Aptiva computers (1999)
      - “*concept*” diffused on CD Microsoft
  - Active
    - The user activates the “dropper” (without knowing it!)
- Incubation phase
  - To ensure the survival of the virus (exception: spy viruses which limit their stay in the environment attacked, by disinfecting themselves after their finished their attack)
    - To limit its detection by the user
    - To limit its detection by the anti-virus

7.3 Life-phases of a virus (2/3)

- Disease phase: the final load will be activated
  - At the head of the code: final load carried out before any infection
  - At the end of the code: final load carried out after the process of infection
  - In the middle of the code: if conditioned by the success or not of the infection
  - Differed release, ex: logic bomb
    - Date system (virus “Friday 13”, CIH)
    - Type particular sequences (112 times “Ctrl+Alt+Del”)
    - A number of openings of a Word document (virus “Colors” after 300 openings of documents)

7.3 Life-phases of a virus (3/3)

- Phase of disease
  - Charge of non-lethal nature
    - Posting images or animations
    - Sounds
  - Lethal loads
    - Attack the data confidentiality
    - Integrity of the system or the data
      - Formatting hard disk
      - Destruction, random modification of the data
      - Availability of the system (random restarting, saturation, simulation of breakdowns of peripherals)
      - Incrimination of the users (introduction of compromising data, use of the system with punishable or criminal purposes)
  - hardware Destruction
    - Theoretically impossible but
    - Possible Destruction of physically stored software (stored in hard)) (ex: BIOS => hardware attacks simulated)
    - Destruction of hard disks or other hardware elements by “accelerated wear” => program requesting these resources considerably, for example
      - Often undetectable by the anti-virus
      - “Spectacular” consequences of their action non visible => seen as a “random” breakdown of components

7.3 Capacities of the viruses to fight and destroy the protections installation

- Ex: Polymorphism (several forms)
  - N.B.: The anti-viruses often function on detection, search for viral signatures
  - The goal of polymorphism is to vary, of copy in viral copy, any fixed element being able to be exploited by the antivirus to identify the virus (set of instructions, in particular character strings)
  - Techniques of polymorphism
    - Rewriting of the code by use of equivalent code (ex following slide)
    - Use of techniques of basic coding on whole or part of the virus or the worm
      - It is a question of varying coding with each infection so that the signature of the virus is each time different

7.3 Capacities of the viruses to fight and destroy the protections installation

- Example in assembly language

```
loc_401010:
   cmp ecx,0
   jz short loc_40101C
   sub byte ptr [eax], 30h
   inc eax
   dec ecx
   jmp short loc_401010
```

```
loc_401010:
   cmp ecx,0
   jz short loc_40101C
   add byte ptr [eax], <random
     value>
   sub byte ptr [eax], 30h
   sub byte ptr [eax], <same
     random value>
   inc eax
   or eax, eax
   dec ecx
   add ecx,0
   jmp short loc_401010
```

7.3 Classification of viruses and worms

- Nomenclatures of virus according to various criteria
  - Format: virus for executable files or documents
  - Target body: virus for boot sector, peripherals drivers
  - Programming language: assembler, source code, interpreted language
  - Behavior: armoured viruses, slow or fast viruses, retroviruses, resident viruses, polymorphic or furtive viruses
  - Nature of the final load: spies viruses, destroying viruses
  - Operating mode: binary viruses, psychological viruses

7.3 Virus for executable files

- The infection relates to a binary target starting from an infected binary file (in general developed out in assembler)
- Take Into account various executable formats
  - .COM or .EXE, 32 bits Windows binaries
  - Files of drivers (virus of peripherals drivers)
  - ELF format Files in Unix

7.3 Types of files being able to contain documents viruses

Caution with Office suite very widespread and so very vulnerable!

Format	Extensions
Scripts WSH	VBS, JS, VBE, JSE, WSF, WSH
Word	DOC, DOT, WBK, DOCHTML
Excel	XLS, XL?, SLK, XLSHTML
Powerpoint	PPT, POT, PPS, PPA, PWZ, PPTHTML, POTHTML
Access	MDB, MD?, MA?, MDBHTML
RTF	RTF
Shell Scrap	SHS
HTML	HTML, HTM, …
XHTML	XHTML, XHT
XML	XML, XSL
MHTML	MHT, MHTML
Adobe Acrobat	PDF
Postscript	PS, EPS, …
Tex/Latex	TEX

7.3 Other types of virus

- Slow viruses: infect only the modified or created executable files (which is a not very frequent event) (ex: Dark Vader)
- Fast viruses: infect all the files carried out or opened, thus work at the same time as the antivirus (ex: Vacsina, Yankee, Dark Avenger, Ithaqua)
- Multi-party or multimode viruses: several types of targets are infected at the same time, for example hard disk starting sectors and executable files (ex: CrazyEddie, Wogob, Nuclear/Pacific)
- Multi-format Virus : able to infect formats belonging to different operating systems (ex: Winux/Lindose able to infect at the same time executable files with the Linux/Unix ELF format and Windows EP format
- Viral kits of constructions: software allowing the automatic creation of virus… (currently all detected) (ex: “The virus lab creation”, generator of worm “VBS Worm Generator”)
- “Psychological Viruses”: bad information sent to a user, by techniques of social engineering, to produce effects equivalent to that of a virus or a worm
  - Emission of e-mails in chain
  - Messages indicating that the existence of a system file (ex: kernel32.dll) is a virus to be eliminated

7.4 Worms

- Self-reproductive programs
- Particular subclass of viruses, able to propagate the infection through a network
- Same algorithms as the viruses
- A worm can be propagated very quickly on the whole planet

7.4 Types of worms

- Simple worms
  - Exploit Generally software failures which allow the running of programs on a distant machine
- Macro-worms
  - Hybrid Programs: both a virus (infection of support transmitted by network) and a worm (use of the network for the transmission), often propagated by attached files containing of the infected documents
- E-mail worms

7.5 Antiviral fight

- Antiviral techniques (anti-virus)
  - Static Mode (analyzes on release of the user)
    - Search for signatures (suite of bits characteristic of a given virus)
    - Spectral analysis: to establish the list of instructions of a program (the spectrum) and look for instructions which are not very used in classical programs and more usual in viruses
    - heuristic Analysis : study of the behavior of a program in order to detect viral behaviors
    - integrity Control : monitoring of the modification of sensitive files (executable, documents)

7.5 Antiviral fight

- Dynamic Mode (resident, analyzes permanently the files and executable)
  - Behavioral monitoring
    - antivirus diverts interruptions towards its profit (ex: 13H or 21 H) and tries to detect any suspect behavior
      - ” attempt to open/write executable files
      - ” Writing on the system sectors
      - ” attempt to be stored as a resident program
  - code Emulation
    - Fight against polymorphic viruses (simulation of the behavior)

7.5 Antiviral fight

- Signature analysis (or scanning techniques)
  - Databases of known viruses
  - Regular update
- Behaviour analysis
  - « Quality » of the detection algorithm
  - Difficulty of calibration

7. Virus: references

- Les virus informatiques : théorie, pratique et applications, Eric Filiol, 2004, Springer
- F. Halsall – Computer networking and the internet – Addison Welseley, 2005 + additional student support at www.pearsoned.co.uk/halsall
- E. Cole, R. Krutz, JW Conley - Network security bible – Wiley, 2005.
- www-rocq.inria.fr/codes/Eric.Filiol/index.html
- www.sophos.com
- www.fsecure.com
- www.viruslist.com
- antivirus-france.com
- www.clusif.asso.fr/index.asp (rubrique infovirus)
- http://www.inoculer.com/